

Textgenetische Prozesse **in** **Digitalen Editionen**

Inaugural-Dissertation
zur Erlangung des Doktorgrades
der Philosophischen Fakultät
der Universität zu Köln
im Fach Informationsverarbeitung

VON
JÖRG HÖRNSCHEMEYER

KÖLN
2013

Erstgutachter: Prof. Dr. Manfred Thaller
Zweitgutachter: Prof. Dr. Reinhard Förtsch
Datum der Disputation: 29. Januar 2014

Abstract

Begünstigt durch den digitalen Medienwandel entstehen zur Zeit neue Editionsformen, deren Nutzen sich vor allem aus der Überwindung medienbedingter Limitierungen gedruckter Editionen ergibt. Der Wegfall dieser einschränkenden Faktoren, wie ein fest vorgegebenes Seitenformat oder die Begrenzung auf eine bestimmte Seitenanzahl und die daraus resultierenden Möglichkeiten, im Prinzip unbegrenzte Mengen an Transkriptionen, Abbildungen und Kontextmaterialien in die Digitale Edition integrieren zu können, sind ein wesentliches Merkmal dieser neueren Ansätze. Trotz dieser Entwicklungen fehlt es im Bereich Digitaler Textgenetischer Editionen bis heute aber weiterhin an geeigneten Methoden und Werkzeugen, um den Entstehungsprozess eines Textes intuitiv nachvollziehen und in geeigneter Weise visualisieren zu können. Besonders zwei Aspekte werden in diesem Zusammenhang bislang vernachlässigt. Zum einen die räumliche Anordnung textgenetischer Prozesse, die vor allem bei der Transkription von Manuskripten nicht in ausreichendem Maße berücksichtigt werden und zum anderen die zeitliche Abfolge dieser Prozesse. Die Vorliegende Arbeit soll diese Bereiche gezielt in den Blick nehmen. Dabei geht es zum einen um eine kritische Analyse der bisherigen Kodierungspraxis solcher Phänomene und zum anderen um die Möglichkeiten, wie solche Prozesse durch browserbasierte Lösungen visualisiert werden können. Die Arbeit ist in einen theoretischen und einen praktischen Teil untergliedert. Der erste Teil gibt einen Überblick über die Entwicklungen im Bereich Digitaler Editionen vor allem seit der Etablierung des WWW als generisch digitales Editionsmedium. Darauf folgt ein kurzer historischer Exkurs über Editionspraktiken zu Zeiten der Druckkultur. Anschließend werden zentrale Problembereiche digitaler textgenetischer Editionen thematisiert und modellhafte Lösungsansätze entwickelt. Abschließend werden diese in einem praxisnahen Teil in Form von beispielhaft implementierten Softwarekomponenten veranschaulicht.

Inhaltsverzeichnis

Einleitung	1
1 Digitale Editionen	5
1.1 Digitale Edition/Begriffsbestimmung	5
1.2 Entwicklung	12
1.2.1 Computer als Hilfsmittel	13
1.2.2 Datenmodellierung	15
1.2.3 Publikation	28
1.2.3.1 Externe Speichermedien	28
1.2.3.2 WWW	30
1.2.4 Software	35
1.2.4.1 Bildbearbeitung	35
1.2.4.2 Texterkennung	36
1.2.4.3 Kollationierung	39
1.2.4.4 Datenerfassung/-bearbeitung	43
2 Digitale Textgenetische Editionen	49
2.1 Exkurs: Textgenese in Druckeditionen	49
2.1.1 Apparate	57
2.1.2 Faksimile	65
2.1.3 Paralleldruck	66
2.2 State-of-the-art Projekte	67
2.3 Erfahrungsbericht	72
2.4 Texttopographie	78
2.4.1 Diplomatische Ansichten	83
2.5 Chronologien und Lesarten	89
2.5.1 Chronologische Abfolgen auf Mikro- und Makroebene	90
2.5.2 Chronologische Lesarten	94
2.6 Synoptische Ansichten	98
2.7 Retrieval	102
2.8 Multiple Hierarchien	104
2.9 Usability	107
3 System Design	111
3.1 Hauptkomponenten	111
3.2 Editor	113
3.2.1 Grundkonzept des Markups	114
3.2.2 Linewidths	115

3.2.3	add-Objekt - Hinzufügen von Textsegmenten	120
3.2.4	del-Objekt - Tilgen von Textsegmenten	121
3.2.5	transpos-Objekt - Transpositionen	123
3.2.6	Modifikationen wiederholen oder zurücknehmen	125
3.2.7	rdg-Objekt - Variantenverzeichnung unterschiedlicher Textzeugen	126
3.2.8	Anlegen von neuen Markup-Elementen und Dateneingabe . . .	126
3.2.9	add- und rootAdd-Objekt	127
3.2.10	Besonderheiten des rdg-Objekts	130
3.2.11	Konfigurationsmöglichkeiten	133
3.2.11.1	Darstellung von Markup-Widgets	134
3.2.11.2	Speicherung unterschiedlicher Sichten	134
3.2.11.3	‘Zitieren’ von Text- und Bildausschnitten	137
3.2.12	Datenformat	138
3.2.13	Serialisierung	141
3.3	SVG-View	144
3.3.1	Browserbasierte Vektorgraphikformate - Evaluation	147
3.3.2	Grundarchitektur	148
3.3.2.1	Polygonlayer	152
3.3.3	Wie kommt der Text ins Digitalisat?	155
3.3.3.1	Manuelles Einfügen	155
3.3.3.2	Automatisches Einfügen	162
3.3.4	Streichungen	165
3.4	Textgenese dynamisch visualisieren	167
3.4.1	Navigationsbaum	167
3.4.2	Chronologische Lesarten hinzufügen	172
3.4.3	line-Widgets für die Variantendarstellung	174
3.4.4	Visualisierung über HTML	175

4 Fazit/Ausblick 179

Literaturverzeichnis 185

Einleitung

*Edition ist die Aufbereitung von Quellen für die wissenschaftliche Benutzung. Ihr Zweck ist die Bereitstellung gesicherter Texte, die einen Rückgriff auf die Überlieferung so weit wie möglich ersetzen sollen.*¹

Diese scheinbar simple Anforderung hat in den letzten 200 Jahren eine Vielzahl von theoretischen Diskussionen über geeignete Editionsmethoden hervorgebracht, die schliesslich zu einer fast unüberschaubaren Zahl an Editionsmodellen geführt haben. Die anfängliche Hoffnung, ein umfassendes epochen- und fächerübergreifendes Modell für sämtliche Textgattungen entwickeln zu können, konnten allerdings nicht erfüllt werden. Statt dessen entwickelten sich in den verschiedenen geisteswissenschaftlichen Disziplinen die unterschiedlichsten Methoden zur kritischen Bearbeitung historisch überlieferter Texte. Vielfältige Überlieferungsumstände, Fragestellungen und länder- bzw. fächerspezifische (Kultur)Techniken führten zu einer starken Diversifizierung dieser Methoden. Somit entwickelte sich die Editionswissenschaft auch wortwörtlich zu einer *Wissenschaft für sich*. Die komplexen Erarbeitungs- und vor allem Darstellungsmethoden in ihren zahlreichen Apparatmodellen und Anhängen waren teilweise so hochspezialisiert und komplex, dass sie dem eigentlichen Sinn, die Texte einem breiteren (Fach)Publikum zugänglich zu machen, nicht mehr gerecht werden konnten. Dabei war die Erkenntnis, einen Text nicht mehr nur als eine streng lineare Folge von Zeichen zu betrachten, bereits wegweisend. Das Problem bestand jedoch darin, dass eine klassische Druckedition, bedingt durch ihre naturgemäss begrenzten Möglichkeiten, nicht in der Lage war, diese editionstheoretischen Anforderungen adäquat umzusetzen. Schlimmer noch, die limitierenden Faktoren bewirkten, dass sich die Methode dem Medium anpasste, und nicht umgekehrt. In diesem Kontext spielten und spielen die Analyse und Darstellung von genetischen Prozessen eine zentrale Rolle. Dies lässt sich allein schon daran ablesen, wie viel Zeit in der klassischen Editions-wissenschaft darauf verwendet wurde, immer aussagekräftigere Apparatmodelle für die Darstellung textgenetischer Prozesse zu entwickeln (siehe Abschnitt 2.1.1). Aber auch innerhalb der elektronischen Medien stellen sich Fragen nach der bestmöglichen Analysier- und Visualisierbarkeit von solchen Informationen. Zum einen unterscheiden sie sich gar nicht so sehr von denen der Druckkultur, da ja auch elektronische Medien durch ihre Ausgabegeräte an einen bestimmten Raum für die Darstellung von Informationen gebunden sind, zum anderen jedoch sehr, da es durch den Wegfall verschiedener Faktoren, wie etwa die Begrenzung des Speicherplatzes oder durch dynamische Visualisierungsmöglichkeiten nun theoretisch möglich ist, alle verfügbaren

¹ Patrick SAHLE: Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels, Bd. 1 Das typographische Erbe, Norderstedt 2013, S. 63.

Textzeugen einer Überlieferung inklusive ihrer Faksimiles und Kontextmaterialien in digitaler Form zur Verfügung zu stellen und diese so miteinander in Beziehung zu setzen, dass ihre komplette Entstehungsgeschichte nachvollziehbar wird. Dabei spielen geeignete Konzepte der Informationsvermittlung wie Parallelarstellung von Textzeugen, Vernetzungsvisualisierungen, Navigationsmechanismen oder das sinnvolle Reduzieren von 'unwichtigen' Informationen eine zunehmend größer werdende Rolle.

Vor allem im Bereich digitaler textgenetischer Editionen ist es mit der bloßen Bereitstellung der Bild- und Textdaten allein aber nicht getan. Hier fehlt es immer noch an geeigneten Methoden und Werkzeugen, die den Entstehungsprozess eines Textes visuell nachvollziehbar machen, und zwar so, dass Transkription und Digitalisat nicht mehr nur separat, unverknüpft nebeneinander präsentiert werden, sondern so, dass sie gegenseitig Bezug aufeinander nehmen und damit die Transkription wieder näher an das eigentliche Forschungsobjekt, den Überlieferungsträger, rücken.² Besonders zwei Komponenten werden in diesem Zusammenhang bislang vernachlässigt. Zum einen die räumlichen Aspekte textgenetischer Prozesse, die vor allem bei der Transkription von Manuskripten nicht in ausreichendem Maße berücksichtigt werden und zum anderen die chronologischen Aspekte, die zu unterschiedlichen Kombinationen und Abfolgen von genetischen Modifikationsprozessen führen können. Darüber hinaus soll aufgezeigt werden, dass in genetischen Editionen vor allem die zahlreichen physischen und logischen Informationsschichten zu Problemen bei der Kodierung mit gängigen OHCO-basierten Markuptechnologien führen und dass alternative Modelle, die z.B. auf stand-off-Methoden beruhen, hier Abhilfe schaffen können. Schließlich sollen diese Konzepte in einem Editionstool zur Kodierung und Visualisierung textgenetischer Prozesse umgesetzt werden. Dieses System ist so konzipiert, dass es in kollaborativen browserbasierten Entwicklungsumgebungen nach dem Vorbild von SaaS-Anwendungen (Software as a service) eingesetzt werden kann. Hier gilt es vor allem zu zeigen, dass mit den neuen technischen Möglichkeiten von HTML-5, Canvas, SVG und javascript im Prinzip alle Komponenten einer klassischen Editions-Desktopanwendung auch in einer clientseitigen Browserapplikation realisiert werden können. Dies betrifft vor allem den Transkriptionseditor, der das Markup mit Hilfe von konfigurierbaren Line-Widgets unterhalb der jeweiligen Zeile einblendet und somit die Usability bei der Annotation des Textes verbessert und beliebige überlappende Strukturen kodierbar macht, das Diplomatiktool, mit dem der Text nicht nur mit der Transkription im Sinne eines Image-Linking-Mechanismus verknüpft, sondern auch wahlweise manuell oder maschinell (über Wort- und Zeilen-Segmentierungsalgorithmen) topographisch exakt auf dem Faksimile positioniert werden kann und ein Annotationswerkzeug, mit dem alternative chronologische Abfolgen kodiert werden können.

² Wie dies ja schon seit geraumer Zeit in der Editionswissenschaft gefordert wird: „Als einzig autoritatives, stabiles und verlässliches Element [bleibt] die Überlieferung. Schon seit längerem geht es auch in der Edition kaum noch um 'Intentionen' und dafür mehr um 'Authentizität'. Diese Authentizität aber kann keinem abstrakten Gegenstand – z.B. einem intendierten linguistischen Code jenseits der Medien – mehr zukommen, sondern allein den physisch vorhandenen Dokumenten der Überlieferung.“ Patrick SAHLE: *Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels*, Bd. 2 Befunde, Theorie und Methodik, Norderstedt 2013, S. 139.

Die Arbeit ist in einen theoretischen und einen praktischen Teil untergliedert. Der erste Teil soll einen Überblick über die Entwicklungen im Bereich Digitaler Editionen vor allem seit der Etablierung des WWW als generisch digitales Editionsmedium geben (Kapitel 1). Darauf folgt ein kurzer historischer Exkurs über Editionspraktiken zu Zeiten der Druckkultur (Abschnitt 2.1). Im Anschluss werden zentrale Problem-bereiche digitaler textgenetischer Editionen thematisiert und in Form von Konzepten und Modellen Lösungsansätze entwickelt. Im praktischen Teil sollen die Implementierung und Beschreibung der verschiedenen Softwarekomponenten im Mittelpunkt stehen (Kapitel 3). Abschließen wird die Arbeit ein kurzes Fazit und ein Ausblick auf zukünftige Herausforderungen, denen sich Entwickler von Informationssystemen innerhalb der Digital Humanities stellen werden müssen.

1 Digitale Editionen

1.1 Digitale Edition/Begriffsbestimmung

Bis heute mangelt es an einer eindeutigen Definition, was eine Digitale Edition eigentlich ist oder was für Bedingungen erfüllt sein müssen, um von einer Digitalen Edition sprechen zu können. Es existieren zwar unterschiedliche Ansätze einer Begriffsbestimmung, es besteht aber weiterhin keine verbindliche Norm. Die Gründe dafür, dass sich noch keine konsensfähige Definition herausgebildet hat, sind recht unterschiedlich. Zum einen sind es die gleichen Diskussionen, die schon zu Zeiten der gedruckten Editionen geführt wurden, zum anderen sind es bestimmte Erwartungshaltungen, die an die neueren digitalen Medien und ihre Möglichkeiten geknüpft werden. Eine Ebene der Begriffsbestimmung zielt auf ihre 'mediale Präsenz' ab. So kann die Definition von der Digitalen Edition als Gegenstand, der sowohl inhaltlich in digitaler Form vorliegt, als auch bei der Publikation konsequent auf digitale Medien setzt,³ nur als erste grobe Einteilung verstanden werden, mit deren Hilfe sich ein Ausschlusskriterium anwenden lässt, das alle anderen Arten von Editionen ausklammert, die eine der beiden Bedingungen nicht erfüllen, also alle gedruckten Editionen etwa unabhängig davon, wie intensiv bestimmte Softwaresysteme zu ihrer Erstellung beigetragen haben. Diese Definition lässt allerdings (zu) viel Spielraum für digital vorliegende und digital publizierte Überlieferungsträger, die man vielleicht nicht direkt als Digitale Editionen bezeichnen würde. Es stellt sich z.B. die Frage, ob eine Sammlung von digitalisierten Überlieferungen in Form von unkommentierten Bitmap-Graphiken schon als Faksimile-Edition bezeichnet werden kann? Das Kriterium der medialen Präsenz alleine reicht hier also nicht aus.

Ein anderer häufig zu findender Ansatz besteht darin, die Digitale Edition mit der Druckedition unter funktionalen Aspekten zu vergleichen, verbunden mit der Anforderung, dass eine Digitale Edition im Prinzip alles das können muss, was eine gedruckte Edition auch kann (und mehr) und zwar in der gleichen (oder in besserer) Qualität: „At their best, so far, most electronic editions do the same as book editions: they just do more of it, perhaps with marginally more convenience. In essence, their product is not significantly different qualitatively to that of heutzutage“⁴ oder auch nach Sahle: „Digitale Editionen müssen den gleichen Ansprüchen genügen wie traditionelle gedruckte Editionen. Sie müssen die bisher erreichte editorische Qualität bewahren

³ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 27.

⁴ Peter ROBINSON: Where We Are with Electronic Scholarly Editions, and Where We Want to Be, in: Jahrbuch für Computerphilologie 5 2004, URL: <http://computerphilologie.digital-humanities.de/jg03/robinson.html>.

und genauso akkurat gearbeitet und zuverlässig sein.“⁵ Es wird gerne darauf hingewiesen, dass die Qualität Digitaler Editionen heutzutage noch starken Schwankungen unterworfen ist. Die Frage nach der Qualität ist aber nicht unproblematisch, da die Kritik meist aus den Reihen der traditionellen Editionswissenschaft kommt und sich auf Qualitätsmerkmale und -standards gedruckter Editionen bezieht. Zuerst einmal sollte dann aber auch danach gefragt werden, was denn genau die Standards und Qualitätsmerkmale der gedruckten Edition sind und ob sich diese auch eins-zu-eins auf eine neue mediale Umgebung anwenden lassen. Die visuellen und funktionalen Standards der traditionellen Editionswissenschaften haben sich über einen Zeitraum von mehreren Jahrhunderten entwickelt.⁶ Natürlich unterlagen auch ihre Methoden und Theorien fortlaufenden Weiterentwicklungen.⁷ Mit der Etablierung visueller und funktionaler Normen hat es die klassische Editionswissenschaft aber insgesamt auch einfacher gehabt. Aufgrund des linearen, zweidimensionalen Druck-Mediums Buch stellten sich viele Standardisierungsfragen in Bezug auf Funktion und Visualisierung überhaupt nicht. Durch den vorgegebenen Raum der Druckseite war das Repertoire an Möglichkeiten eng begrenzt. Dem eigentlichen transkribierten (und zumeist auch konstituierten) Editionstext folgte meist eine unterschiedliche Anzahl an kritischen Apparaten unterhalb des Textes auf der selben Seite und eine Übersetzung oder ein Faksimile-Abdruck auf der gegenüberliegenden Seite. Damit waren die visuellen Anordnungsmöglichkeiten aber auch schon weitestgehend erschöpft und bildeten eine Art Quasi-Standard, der in nahezu allen Editionsprojekten ähnlich aussah. Die Zugriffsmöglichkeiten auf das Quellenmaterial in Form von Recherchemechanismen und Navigationsstrukturen waren größtenteils genormt. Zwar vielfältig in ihrem Inhalt aber doch in ihrer äußeren Form immer ähnlich, sind die einzigen Suchmechanismen innerhalb der Printedition die Indizes (meist am Ende der gedruckten Edition oder in Extrabänden mit Verweisen auf die Seitenzahl oder die Dokumentnummer).

Diese Art an Zugriffsfunktionen stellt aber nur einen sehr kleinen Teil der Möglichkeiten dar, die dem Benutzer einer Digitalen Edition zur Verfügung stehen kann. Neben den inzwischen schon standardmäßig angebotenen Möglichkeiten von Volltextsuchen, feldspezifischen Suchen oder diversen Trunkierungsmechanismen, Gewichtung von Suchtermen, Umgebungssuchen etc. spielen darüber hinausgehende Funktionen eine immer größere Rolle: Fuzzysearches, Datamining, semantische Suchmöglichkeiten (Stichwort Semantic Web, AI), 'intelligente' Rankingmechanismen etc. Neben den

⁵ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 204.

⁶ Und dies auch nicht linear in eine Richtung sondern beeinflusst durch unterschiedliche Schulen und nationale Traditionen. In Frankreich auf andere Weise als in der philologisch geprägten Editionswissenschaft in Deutschland oder der eher dokumentarischen Editionswissenschaft des angloamerikanischen Raumes. Auf die beiden ersteren wird im Verlaufe der Arbeit noch näher eingegangen werden, für letztere sie z.B. Peter SHILLINGSBURG: Anglo-amerikanische Editions-wissenschaft. Ein knapper Überblick, in: Text und Edition – Positionen und Perspektiven 2000, hrsg. v. Rüdiger NUTT-KOFOTH/Bodo PLACHTA/Hermann van VLIET H.T.M. und Zwerschina, S. 143–164 oder DERS.: 'The subject of our Mirth'. The Aesthetic Object in Anglo-American Editing, in: Perspectives of Scholarly Editing / Perspektiven der Textedition. Hg. von H.T.M. van Vliet und Bodo Plachta. 2002, S. 97–112.

⁷ Vgl. etwa die Diskussion über die zahlreichen Entwicklungen im Bereich der kritischen Apparate in der deutschen Editionswissenschaft Kapitel 2.1.1.

vielfältigen Möglichkeiten des Suchens und Navigierens (und damit auch unmittelbar verbunden) stellt sich z.B. zusätzlich die Frage, welches Kontextmaterial zu einer 'guten' Digitalen Edition gehört. Der im Prinzip unbegrenzte Speicherplatz ermöglicht es ja nun, sämtliches Zusatzmaterial, auf das in der Printedition meist nur in einer Fußnote verwiesen wurde zu veröffentlichen oder zu verlinken. Gehören (abgesehen vom Digitalisat) auch weiterführendes Bild- und Tonmaterial dazu? Wie sieht es mit der Verknüpfung zu thematisch verwandten Ressourcen aus? Was soll referenziert werden und wie soll das geschehen? Gehören Artikel aus einschlägigen Online-Lexika dazu? Personenatenverknüpfungen via GND? Darf der Zeitungsartikel über den Untergang des Raddampfers Erie⁸ auf der Homepage der 'Maritime History of the Great Lakes' bei einer Edition der Balladen Theodor Fontanes nicht fehlen? Gehört die Vertonung von 'John Maynard' durch den deutschen Komponisten Achim Reichel auf youtube dazu?⁹ Was ist unter vertretbarem Aufwand machbar? Was ist sinnvoll? Was ist finanzierbar? Diese Fragen muss jedes Editionsprojekt selbst für sich beantworten.

Ebenso fragwürdig erscheint es, die visuellen Standards des Drucks als qualitative Norm für visuelle Standards in einem digitalen Medium zu erheben. Der Vorteil Digitaler Editionen liegt nicht zuletzt darin, dass Textvarianten oder genetische Prozesse viel anschaulicher dargestellt werden können, als dieses im Druck möglich ist.¹⁰ Im Prinzip können in einer Digitalen Edition alle Repräsentationen eines Textes in beliebiger Kombination nebeneinander dargestellt werden, seien es die vorhandenen Varianten, Entwürfe, Fassungen, digitalisierte Faksimile, Transkriptionen, Übersetzungen, Lesetexte, diplomatische Abdrucke etc. Diese wiederum können in beliebigen synoptischen Vergleichsmodi auf der Basis von Worten, Zeilen, Abschnitten, Seiten oder auf der Dokumentenebene gegenübergestellt werden.

Die Ursachen, die zu dem Eindruck der mangelnden Qualität der Digitalen Editionen führen, liegen sicherlich zum Teil in der unzureichenden Herausbildung solcher funktionalen und visuellen Standards begründet. Derzeit sieht es so aus, dass viele Projekte ihre ganz eigenen individuellen Maßstäbe setzen, was erst einmal positiv zu bewerten ist, da ja gerade die Erprobung neuer innovativer Techniken zu neuen Standards führen kann. Hier befindet sich das relativ junge Feld der digitalen Editions wissenschaft aber auch weiterhin noch in einer Testphase. Ein anderes großes Problem liegt im Bereich der Qualitätssicherung und geht oft einher mit einer fehlen-

⁸ <http://images.maritimehistoryofthegreatlakes.ca/63668/data?n=9>

⁹ In der Tat existieren solche Ansätze. Die Edition „Uncle Tom's Cabin and American Culture“ etwa verweist auf umfangreiches Tonmaterial: „The archive includes playable versions of most of the hymns and spirituals quoted in Uncle Tom's Cabin.“ Beecher-Stowe, Harriet: Uncle Tom's Cabin and American Culture. Directed by Stephen Railton, Charlottesville, University of Virginia, 1998-2007. Online: <<http://www.iath.virginia.edu/utc/>>.

¹⁰ Es sind zahlreiche Aussagen darüber zu finden, wie beschränkt die Möglichkeiten in dieser Hinsicht bei einer gedruckten Edition sind. Stellvertretend hierfür sei auf die Aussage von Walter Fanta über die praktische Unmöglichkeit der synoptischen Präsentation des Musil-Nachlaß in Druckform verwiesen: „Die Blätter der beiden Manuskriptgruppen mit ihrem komplexen bzw. inkohärenten Seitenaufbau sind extrem schwer zu edieren, eine synoptische Darstellung von Faksimile und Transkription auf Papier, das editorische Ideal des Kafka-Herausgebers Roland Reuß, ist für sie selbst heute noch praktisch unerfüllbar.“ Walter FANTA: Zur Immortalität elektronischer Korpora am Beispiel der Musil-Edition, in: Jahrbuch für Computerphilologie (11 noch unveröffentlicht) 2011, URL: <http://computerphilologie.digital-humanities.de/jg09/fanta.html>, §16.

den institutionellen Anbindung solcher Editionsprojekte. Die finanzielle Förderung ist meist begrenzt auf einen gewissen Zeitraum. Im Idealfall findet sich noch eine Institution (ein universitäres Rechenzentrum etwa), die bereit ist, nach Projektende das Hosting der Edition zu übernehmen. Was aber noch immer zu wenig Berücksichtigung findet, ist die Notwendigkeit, solche Projekte weiterhin zu pflegen, sie also technisch, visuell und funktional auf dem neuesten Stand zu halten. Dadurch wirken sie oft schon nach wenigen Jahren veraltet, was wiederum zusätzlich den Eindruck von mangelnder Qualität verstärkt.

Die Frage stellt sich hier aber auch generell, ob die Digitale Edition mit den Definitionsmustern klassischer Printeditionen adäquat zu fassen ist. Bei gedruckten Editionen handelt es sich fast immer um eine Auswahl von Überlieferungsträgern, die für den Editor eine bestimmte Relevanz für bestimmte wissenschaftliche Fragestellungen besaß.¹¹ Die Notwendigkeit, eine solche kanonische Vorauswahl zu treffen, wurde nicht zuletzt durch die begrenzte Aufnahmekapazität des Mediums Buch beeinflusst. In Zeiten der digitalen Datenspeicherung fallen solche Gründe natürlich weg und es ist prinzipiell möglich, die komplette Überlieferungsgeschichte durch alle verfügbaren Textzeugen zu erhellen. Natürlich spielen auch hier nach wie vor limitierende Faktoren eine große Rolle, wie die des Arbeitsaufwandes und der verfügbaren finanziellen Ressourcen. Zu Zeiten der Printedition galt gerade die Auswahl der zu edierenden Quellen als eine der zentralen Aufgaben des Editors. Wenn aber gerade diese Funktion der Auswahl wegfällt, dann erfüllt z.B. eine Digitale Archiv-Edition, deren Anliegen es ist, alle Textzeugen verfügbar zu machen, diese zentrale Aufgabe (bewusst) nicht mehr.

Eine Definition, die sich aus einem Vergleich zwischen Druck-Standards und digitalen Standards ableitet, führt also auch nicht wirklich zu einer zufriedenstellenden Definition dessen, was eine Digitale Edition im Kern ausmachen sollte. Es existieren noch zahlreiche weitere Ansätze und Forderungen bezüglich dessen, was eine Digitale Edition sein sollte. Diese sind häufig beeinflusst durch Methoden, Theorien, Entwicklungen und Vorlieben der einzelnen Fachdisziplinen der Geisteswissenschaften und setzen ihre Schwerpunkte meist unterschiedlich. Oft sind sie allgemeinerer Natur, betreffen aber auch konkrete technische Aspekte. Nach Robinson¹² umfassen die Grundanforderungen an eine Digitale Edition im Prinzip diejenigen, die auch schon in der klassischen Editionstheorie die Basis bildeten: Erschließung, Wiedergabe, Nutzbarkeit,¹³ allerdings ergänzt um zwei wesentliche Aspekte: „A critical digital edition may present an edited text, among all the texts it offers“¹⁴ und „A critical digital

¹¹ Dieses Vorgehen an sich ist natürlich schon kritisch zu hinterfragen, da der Editor mit der Kanonisierung der Textzeugen ein bestimmtes Erkenntnisinteresse des Publikums voraussetzt, das so aber nicht unbedingt vorhanden sein muss. Der wissenschaftliche Wert der Edition ist somit nur von begrenzter Dauer, und es bleibt zu vermuten, dass neue Fragestellungen sich mit einer anderen Quellenauswahl besser beantworten ließen.

¹² Peter ROBINSON: What is an Electronic Critical Edition?, in: *Variants: The Journal of the European Society for Textual Scholarship* 1 2002, S. 51–57.

¹³ Vgl. Scheibe: „Die Hauptaufgabe einer Edition, besonders aber einer wissenschaftlichen Edition, ist es, das zu einem Werk vorliegende Material zu sammeln, zu sichten, zu analysieren, zu bewerten und wiederzugeben.“ Siegfried SCHEIBE: Welche Editionsart für welchen Zweck?, in: *Berliner Beiträge zur Editionswissenschaft* 3 (1998), S. 43–61, hier S. 48.

¹⁴ ROBINSON: What is (wie Anm. 12).

edition allows space and tools for readers to develop their own hypotheses and ways of reading“¹⁵, was schon auf die zwei wesentlichen Faktoren hindeutet, dass 1. ein nach traditionellen Maßstäben konstituierter Text nun nicht mehr das oberste Ziel sein muss, sondern nur noch eine mögliche Präsentation unter vielen und 2., dass dem Benutzer der Edition mehr Raum und Möglichkeiten für eigene Analysen eingeräumt werden sollten.¹⁶

Weniger theoretische, mehr am konkreten Gegenstand orientierte Definitionen sind bei Tanselle und Shillingsburg zu finden. Für sie besteht die ideale Digitale Edition aus 1. einem Faksimile des Originals und 2. einer exakten vollständigen Transkription.¹⁷ Ergänzt werden die Anforderungen von Shillingsburg durch 3. ein Referenzsystem, das alle Varianten und Kontextinformationen verknüpft und 4. ein Navigationssystem, das den Benutzer durch die Edition führt.¹⁸ In der Diskussion um die ideale Digitale Edition existiert noch eine große Anzahl weiterer listenartiger Anforderungsprofile,¹⁹ die so oder so ähnlich formuliert die ideale Vorstellung einer Digitalen Edition widerspiegeln, sich in der Praxis aber allzu häufig aus verschiedenen Gründen nicht als praktikabel erweisen. Hier muss z.B. bemerkt werden, dass es aus finanziellen oder urheberrechtlichen Gründen nicht jedem Editionsvorhaben möglich ist, Faksimileabbildungen des Originals zur Verfügung zu stellen. Allzu oft besitzen Archive die Rechte am Bildmaterial und treten diese auch nicht an die jeweiligen Editionsprojekte ab.²⁰

Aber nicht nur Diskussionen um Definitionen und Anforderungen, was eine Digitale Edition sein soll und was sie zu leisten hat, beeinflussten die Debatten, sondern auch Diskussionen um eine sich verändernde Editionspraxis innerhalb des digitalen Mediums.²¹ Ein grundsätzlicher Ansatz betrifft die Diskussion um die unterschiedliche Gewichtung der Erschließungstiefe und der Erschließungsbreite von Überlieferungen

¹⁵ Ebd.

¹⁶ Auch zu finden bei Monella: „1. a digital critical edition allows the reader to verify and call into question the work of the editor. 2.it builds up an ‘open’ model of the text, not implying that the text created by the editor is the text.“ Paolo MONELLA: Towards a digital model to edit the different paratextuality levels within a textual tradition, in: Digital Medievalist 4 2008, URL: <http://www.digitalmedievalist.org/journal/4/monella/>, §5.

¹⁷ Vgl. G. Thomas TANSELLE: Textual Instability and Editorial Idealism, in: Studies in Bibliography 49 1996, S. 1–60, URL: <http://etext.lib.virginia.edu/bsuva/sb/> und Peter SHILLINGSBURG: Principles for Electronic Archives, Scholarly Editions, and Tutorials, in: The Literary Text in the Digital Age. Richard J. Finneran (Hrsg.) S. 23–35.

¹⁸ Ebd.

¹⁹ Vgl. z.B. Edward VANHOUTTE: Prose Fiction and Modern Manuscripts: Limitations and Possibilities of Text-Encoding for Electronic Editions, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD/Katherine O'BRIEN O'KEEFE/John UNSWORTH, S. 161–180.

²⁰ Dies auch aus teilweise nachvollziehbaren Gründen, sind ja auch die Archive darauf angewiesen, ihre eigenen Digitalisierungsprojekte zu refinanzieren und durch nachweisbare Nutzerzahlen ihre Bemühungen in dieser Richtung zu rechtfertigen.

²¹ Vgl. Roland S. KAMZELAK: Hypermedia – Brauchen wir eine neue Editionswissenschaft?, in: Beihefte zu Editio, Bd 12: Computergestützte Text-Editionen 1999, S. 119–126 Vanhoutte geht hier noch einen Schritt weiter und argumentiert, dass die veränderten medialen Bedingungen nicht nur zu einer neuen Editionspraxis geführt haben, sondern die Theroiebildung der Editionswissenschaften nachhaltig beeinflussten. Edward VANHOUTTE: Traditional editorial standards and the digital edition, in: Proceedings of the Emblem Project Utrecht Conference on Dutch Love Emblems and the Internet (November 2006) 2007, S. 157–174, URL: http://emblems.let.uu.nl/static/images/project/learned_love_157-174.pdf.

und mit ihr unmittelbar verbunden die Diskussionen um die kritische und unkritische (oder auch vorkritische) Erschließung. Die Befürworter der breiten, dokumentarischen Erschließung sehen die Vorteile dieser Praxis vor allem in der Verfügbarmachung einer umfassenden Quellenbasis, die auch jene Dokumente beinhaltet, die sonst nicht in den Fokus des wissenschaftlichen Interesses rücken würden,²² der Möglichkeit, auch umfassendes Kontextmaterial zur Verfügung zu stellen²³ und dem Mehrwert, durch Möglichkeiten der Vernetzung aus all diesen Quellen und Materialien neue wissenschaftliche Forschungskorpora zu erschließen.²⁴ Die Vertreter der Gegenposition, die Befürworter einer weit gehenden und exakten Tiefenerschließung (beispielsweise durch die Verwendung umfangreicher semantischer Markup-Strukturen) sehen hingegen die Vorteile in den zahlreichen Verarbeitungs- und Analysemöglichkeiten, die erst durch eine solche Form der Erschließung möglich werden.²⁵ Es gibt auch kritische Stimmen, die vor einer zu großen Überlieferungsvielfalt warnen, wie etwa Shillingsburg: „Electronic scholarly editions become electronic archives which may store all documentary sources both in searchable text and digital images, but they may suffer from the pitfall of becoming chaotic collections of material.“²⁶ Eine Vereinigung der beiden vermeintlich gegenläufigen Tendenzen beschreibt Stuart Jenks in seinem Pyramidenmodell des Digitalen Edierens.²⁷ Nach Jenks durchläuft der optimale Prozess des Edierens im digitalen Medium zuerst eine horizontale Ebene der breiten, dokumentnahen Erschließung, der mehrere vertikale Ebenen der Tiefenerschließung folgen können. Diese Pyramidenstruktur beschreibt den Vorteil Digitaler Editionen, den Arbeitsprozess in einzelne Teilprozesse gliedern und verschiedene Phasen der Erschließung trennen zu können. Auf eine erste möglichst breite Erschließung durch dokumentnahe Abbildungen und Basistranskription können verschiedene Phasen der textkritischen Tiefenerschließung folgen. Auch Edward Vanhoutte erkennt das Problem und spricht auf der einen Seite von einer Archivfunktion, die die ganze Breite der Überlieferung zu berücksichtigen hat, gleichzeitig aber auch von einer Museumsfunktion, die die Bedürfnisse der Tiefenanalyse unterstützen sollte.²⁸ Auch Sahle sieht schließlich das ganze Potential Digitaler Editionen erst in der Kombination der beiden Ansätze ausgeschöpft:

²² Vgl. Gibbs: „A greater focus on preserving quick and dirty transcription will provide a valuable complement to canonical editions and make available more versions of manuscripts that actually existed, as well as texts that would never get a critical edition in the first place.“ Fred GIBBS: New Textual Traditions from Community Transcription, in: Digital Medievalist 7 2011, URL: <http://www.digitalmedievalist.org/journal/7/gibbs/>, §31.

²³ Vgl. Peter ROBINSON: ... but what kind of electronic editions should we be making?, in: ALLC-ACH 96 Conference Abstracts 1996, S. 81–82, URL: <http://helmer.aksis.uib.no/allc/finneran.pdf>, hier S. 81.

²⁴ „Invisibility and disconnectedness between manuscripts constrains our ability to build new research corpora“ GIBBS: Community Transcription (wie Anm. 22), §1.

²⁵ Vgl. SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 229.

²⁶ Peter SHILLINGSBURG: Scholarly Editing in the Computer Age. (3. Auflage), Ann Arbor 1996, S. 165.

²⁷ Vgl. Stuart JENKS: KISS [Keep It Simple, Stupid]: Elektronische Quelleneditionen mit einfachsten Mitteln, in: Quellen und Quelleneditionen im neuen Medienzeitalter 2002, hrsg. v. Manfred THALLER, S. 27–37, 31f.

²⁸ Vgl. Edward VANHOUTTE: Where is the editor? Resistance in the creation of an electronic critical edition, in: Human IT. Tidskrift för studier av IT ur ett humanvetenskapligt perspektiv 1998, URL: <http://etjanst.hb.se/bhs/ith/1-99/ev.htm>.

„Digitale Editionen können ihre Vorteile aber dann am besten ausspielen, wenn sie sowohl in ihrer Tiefe, als auch in ihrer Breite 'umfassend' sind.“²⁹ Dies führt allerdings zu der Problematik, dass Editionen, die nicht die 'nötige' Breite und Tiefe besitzen, der Status der Digitalen Edition streitig gemacht wird: „Eine Grenze lässt sich eher durch den Grad der Erschließung und den Informationsreichtum der recodierenden Wiedergabe bestimmen. Erst ab einer gewissen historischen und/oder philologischen Durchdringung des Stoffes und erst beim Vorliegen einer vollständigen Textwiedergabe wird man von einer Edition sprechen, davor eher von einem Archiv.“³⁰ Hier befindet sich die Digitale Edition zur Zeit in dem Dilemma, allen Ansprüchen, die im digitalen Medium prinzipiell erfüllbar sind, auch gerecht werden zu müssen, um als vollwertige Digitale Edition gelten zu können. Es scheint allerdings sinnvoller zu sein, sich von solchen Maximalvorstellungen nicht allzu sehr leiten zu lassen und auch Gegenpositionen zu vertreten oder besser sich gleich auf Minimalstandards zu einigen.

Bei all den unterschiedlichen Ansätzen ist eine unübersehbare Tendenz zu beobachten, die im Gegensatz zur klassischen Editionspraxis eine dokumentnahe, unkritische, objektive³¹ Basistranskription der Quelle befürwortet, um den Blick auf das Original nicht durch einen nach welchen Gesichtspunkten auch immer konstituierten Text zu verstellen. Anstelle der kritischen Edition soll nach Gibbs die nicht-kritische 'documentary' Edition treten, also das Sichtbarmachen des Manuskripttextes anstatt einer Konstituierung des Textes.³² Lou Burnhard³³ und Allen Renear³⁴ fordern gar eine eigene 'Theorie der unkritischen Edition' mit einer quellennahen Basistranskription. Ähnliche Überlegungen stellt Fanta an. Er schlägt eine Begriffserweiterung vor, die eine Unterscheidung zwischen einem „primären edierten Text (unmittelbare Re-

²⁹ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 111.

³⁰ Ebd., S. 154.

³¹ Auch hierzu werden rege Debatten geführt, inwieweit es sich bei dokumentnaher Kodierung und Transkriptionen von Überlieferungen überhaupt um unkritische oder gar objektive Vorgänge handeln kann. Vgl. hierzu die Überlegungen von Kline über die nicht-kritische aber nicht unkritische Edition Mary-Jo KLINE: A Guide to Documentary Editing, Baltimore 1987, S. 24, Pichler zur Abhängigkeit jeder Transkription vom Auswertungsinteresse Alois PICHLER: Transcriptions, Texts and Interpretation, in: Culture and Value. Beiträge des 18. Internationalen Wittgenstein Symposiums 1995, hrsg. v. Kjell S. JOHANNESSEN/Tore NORDENSTAM, S. 690–695, URL: wab.uib.no/~alouis/pichler-kirchb95a.pdf oder auch Daniel GREENSTEIN/Lou BURNARD: Speaking with One Voice – Encoding Standards and the Prospects for an Integrated Approach to Computing in History, in: Computers and the Humanities 29.2 (1995), S. 137–148, hier S. 139 und SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 198. In diesem Zusammenhang soll mit unkritisch und objektiv die Gegenposition zu jeglicher kritischen Konstituierung eines fiktiven Textes gemeint sein. Vgl. dazu Edward VANHOUTTE: Putting Time back in Manuscripts: Textual Study and Text Encoding with Examples from Modern Manuscripts, in: ALLC/ACH 2002 Abstracts 2002, S. 126–127, URL: <http://ctb.kantl.be/vanhoutte/pub/2002/allc02abstr.htm> und DERS.: Prose (wie Anm. 19).

³² „I am suggesting a shift in values from privileging the critical edition to prioritizing the creation of visible manuscript text.“ GIBBS: Community Transcription (wie Anm. 22), §8.

³³ Lou BURNARD: From two cultures to digital culture: the rise of the digital demotic, in: Informatica Umanistica. Dalla ricerca all'insegnamento 2003, hrsg. v. Domenico FIORMONTE, URL: <http://users.ox.ac.uk/~lou/wip/twocults.html>.

³⁴ Allen RENEAR: Literal Transcription – Can the text ontologist help?, in: New Media and the Humanities: Research and Applications 2001, hrsg. v. Domenico FIORMONTE/Jonathan USHER, S. 23–30.

präsentation der Quelle)³⁵ und einem „sekundären edierten Text (Rekonstruktion, Emendation, Normalisierung, Lektüre-Vermittlung)“³⁶ macht. Das Konzept der dokumentnahen Basistranskription vertritt auch ausdrücklich Robinson.³⁷

Neben den allgemeineren, eher theoretischen Definitionsversuchen führten unterschiedliche Entwicklungen z.B. im Bereich der Markup-Theorien zu sehr exakten Vorstellungen davon, auf welchen technologischen Grundlagen eine Digitale Edition aufgebaut sein sollte. Vor allem die Entwicklung deskriptiver Markup-Sprachen (besonders SGML, XML oder auch die TEI) führte zu der (später noch kritisch zu diskutierenden) Aussage, dass es heutzutage ja Konsens sei, dass eine Edition ein Archiv der Transkriptionen beinhalten müsse, die nach den Richtlinien moderner Datenstandards, wie die der TEI, ausgezeichnet wurden,³⁸ oder wie Price es ausdrückt: „XML, has become the de facto standard for serious humanities computing projects.“³⁹ Um zu verstehen, wie es zu diesen eher fragwürdigen und sich selbst konzeptuell beschränkenden Aussagen kommen konnte, sollen im nächsten Kapitel die unterschiedlichen technologischen Entwicklungen und ihre Auswirkungen auf die heutige Editionspraxis Digitaler Editionen näher untersucht werden.

1.2 Entwicklung

Die Entwicklung Digitaler Editionen ist nicht als ein einfacher linearer Prozess der kontinuierlichen Weiterentwicklung von Methoden und Theorien und deren Überführung in ein neues Medium zu betrachten. Vielmehr wurde sie durch unterschiedliche Entwicklungen aus verschiedenen Bereichen beeinflusst. Das Spektrum der Einflüsse ist dabei recht breit. Zu Beginn (und teilweise auch bis heute noch) geprägt durch Vorstellungen, die sich weiterhin stark den Fragestellungen und Prinzipien der klassischen Druckkultur verpflichtet sahen, wurde der Einsatz des Computers vor allem in den Bereichen gefördert, wo er als Hilfsmittel angesehen wurde, an das eigentliche Ziel, die gedruckte Edition, ressourcensparender zu gelangen. Hier ging es vor allem darum, arbeitsintensive sequenzielle Prozesse wie etwa das Kollationieren unterschiedlicher Textzeugen algorithmisch zu formalisieren und maschinell zu verarbeiten und mit Hilfe von Textverarbeitungsprogrammen und typographischen Satzprogrammen den eigentlichen Druck vorzubereiten.⁴⁰ Mit dem Aufkommen günstiger Speichermedien wie der CD-Rom erschienen erste Editionen dann auch direkt in digitaler Form. Die

³⁵ FANTA: Immortalität (wie Anm. 10), §30.

³⁶ Ebd., §30.

³⁷ Peter ROBINSON: The Canterbury Tales and other Medieval Texts, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD/Katherine O'BRIEN O'KEEFFE/John UNSWORTH, S. 74–91, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/robinson.xml.

³⁸ Vgl. J. Graham W. Osborn D.P. O'DONNELL C. Karkov/R. Rosselli Del TURCO: The Visionary Cross: An Experiment in the Multimedia Edition. (Conference Paper), in: Proceedings of Digital Humanities 2007. Zitiert nach SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 183.

³⁹ Kenneth M. PRICE: Electronic Scholarly Editions. (Teil IV Methodologies: The Text Encoding Initiative and the Study of Literature), hrsg. v. Susan SCHREIBMAN/Ray SIEMENS, Oxford 2008, URL: <http://www.digitalhumanities.org/companionDLS/>, §24.

⁴⁰ Vgl. hierzu z.B. frühe Leitfäden, wie die EDV-Fibel für Editoren. Wilhelm OTT/Hans Walter GABLER/Paul SAPPLER: EDV-Fibel für Editoren, Tübingen 1982.

Entwicklung stabiler, weiträumiger Zeichensätze, deskriptiver Markup-Sprachen und die Etablierung des WWW als Publikationsmedium sorgten dann seit Mitte der 90er Jahre für eine immer schnellere Verbreitung und Standardisierung. Bei den Markup-Systemen spielten vor allem hierarchische baumartige Datenstrukturen eine tragende Rolle. Hier übten SGML, XML und die Richtlinien der TEI einen großen Einfluss auf die Entwicklung standardisierter Auszeichnungsformate aus. Den Vorteil in der Verwendung dieser Markup-Sprachen sah man vor allem darin, implizite Textstrukturen mit Formen expliziter Beschreibung auszeichnen zu können, um eine maschinelle Verarbeitung der Daten zu ermöglichen. Durch die Entwicklung spezialisierter Softwaresysteme versuchte man, den Editor den gesamten Produktionsprozess hindurch zu begleiten, von der Aufnahme der Daten über verschiedene Verfahren der Textanalyse bis hin zur fertigen (Online-)Publikation. Unter dem Schlagwort 'user-driven' rückte auch die Usability auf Seiten der Editionsutzer verstärkt in den Fokus des Interesses. Ebenso zu nennen sind hier die Diskussionen um verteilte kollaborative Arbeitsumgebungen, wie etwa die Entwicklungen im Bereich der sogenannten Virtual Research Environments (VRE). In letzter Zeit vermehrt zu beobachten ist die Tendenz, den Ort der Publikation auch gleichzeitig als Arbeitsumgebung zu nutzen. Browserbasierte Lösungen und kollaborative Webanwendungen spielen hier eine zunehmend größere Rolle.

Im folgenden sollen die zuvor skizzierten Themenfelder im Hinblick auf ihren Einfluss auf die Entwicklungen diskutiert werden. Dabei spielen besonders Fragen eine Rolle, wie nachhaltig diese Entwicklungen den Arbeitsprozess des Editors beeinflusst haben, ob und wenn ja, wie groß das Entwicklungspotential in den jeweiligen Bereichen noch ist und auch welche Entwicklungen eher als kritisch und diskussionswürdig zu betrachten sind. Da der Autor dieser Arbeit Digitale Editionen nach der Definition von Patrick Sahle als „Editionen mit digitalen Inhalten und digitaler Publikationsform“⁴¹ verstanden wissen will, soll nur kurz auf die Frühphase des maschinellen Einsatzes eingegangen werden, deren Ziel in der Regel noch darin bestand, den Computer nur als Werkzeug zur Erstellung der klassischen Printeditionen einzusetzen. Von größerem Interesse sind hier die Entwicklungen, die das Digitale Medium auch als generische Publikationsmöglichkeit betrachten und vor allem seit der Zeit der Etablierung des WWW als Publikationsmedium einen großen Einfluss auf die Editionswissenschaft ausüben.

1.2.1 Computer als Hilfsmittel

Schon sehr früh in der Geschichte der Computertechnologie erkannten Geisteswissenschaftler, wie sie die neue Art der formalisierten, auf mathematischen Prinzipien beruhende Datenverarbeitung für ihre Zwecke nutzen konnten. Eingesetzt wurde sie z.B. dort, wo die rechenintensive Verarbeitung von großen Datenmengen zu einer merklichen Arbeitserleichterung führte, wie etwa beim automatischen Kollationieren von Textzeugen. Die Begeisterung in diesem Bereich war zu Beginn recht groß, versprach man sich doch besonders auf diesem Gebiet große Vorteile gegenüber dem manuellen

⁴¹ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 27.

Textvergleich und man hegte die Hoffnung, nun den gesamten Prozess automatisieren zu können.⁴² Im Laufe der Zeit stellte sich jedoch heraus, dass dies nicht so ohne weiteres möglich war. Der automatische Vergleich von Textzeugen war nicht so trivial wie es anfangs erschien. Was für relativ überschaubare Abweichungen bei weitgehender Zeilenparallelität (wie etwa bei Verstexten) noch recht gut funktionierte, war bei großräumigeren Abweichungen schwieriger zu automatisieren. Eine Vorbereitung der Textzeugen war in vielen Fällen unerlässlich. Textzeugen, die größere Lücken aufwiesen oder nur fragmentarisch erhalten waren, mussten erst mit den zu Varianten Textzeugen parallelisiert werden, indem man gemeinsame Ankerpunkte setzte, um dem Vergleichsalgorithmus die richtigen Textpassagen für die Analyse übergeben zu können, wenn man so will eine erste Form des deskriptiven Markups. Aber auch in anderen Bereichen gewann der Computer immer mehr an Bedeutung und wurde bereits bei der Planung der Projekte fester Bestandteil der Mittelausstattung.⁴³ Klassische Anwendungsfelder waren vor allem die Datenaufnahme, die Speicherung in Rechercsystemen und die Vorbereitung für den eigentlichen Druck. Rechercsysteme wie Notabene, spezialisierte Typographische Satzsysteme wie Tex CET (Critical Edition Typesetter)⁴⁴ oder komplexere Arbeitsumgebungen wie Tustep,⁴⁵ bestimmten immer häufiger die digitale Arbeitsumgebung des Editors. Unterschiedliche Module, wie etwa bei Tustep vorhanden, unterstützen das Kollationieren, das Kompilieren von Apparaten, die Erstellung von Konkordanzlisten oder die Extraktion von Registern und Indizes aus den zu publizierenden Daten. Später nahmen Textverarbeitungssysteme wie etwa der Classical Text Editor⁴⁶ (CTE) das wysiwyg-Prinzip auf und erlaubten dem Editor, sich schon während der Datenaufnahme eine (ungefähre) Vorstellung davon zu machen, wie das publizierte Werk im Druck später aussehen würde. Allen solchen Systemen gemein war, dass die Recherche-, Analyse- und Satzwerkzeuge nur für die Bearbeitungsphase bis zur Druckvorlagenerzeugung gedacht waren. Dennoch konnten diese frühen Werkzeuge auch schon für erste methodische Weiterentwicklungen sorgen. Sahle beschreibt dies beispielhaft für die Stemmatalogie.⁴⁷ Die Umsetzung statistischer und empirischer Methoden, die auf einer algorithmischen Formalisierung bereits angewendeter analoger Verfahren beruhten, wie beim automatischen Kollationieren, führte dazu, dass sich die „Stemmabildung langsam von einem Indizienverfahren zu einem statistischen Verfahren“⁴⁸ entwickelte und somit die Terminologie der 'Abstammung' mit Hilfe der aus diesen Methoden gewonnenen Erkenntnisse durch die Terminologie der 'Verwandtschaft' bzw. 'Ähnlichkeit' ergänzt oder ersetzt werden konnte. Hier führte der Einsatz des Computers in dieser Frühphase also auch

⁴² Susan M. HOCKEY: *Electronic Texts in the Humanities: Principles and Practice*, Oxford 2000, S. 125.

⁴³ FANTA: *Immortalität* (wie Anm. 10), §5.

⁴⁴ Bernt KARASCH: *Critical Edition Typesetter (CET)*. Ein Programm zum Satz textkritischer Editionen auf PCs, in: *Beihefte zu Editio*, Bd 12: *Computergestützte Text-Editionen 1999*, S. 87–99.

⁴⁵ <http://www.tustep.uni-tuebingen.de/>

⁴⁶ <http://cte.oeaw.ac.at/>

⁴⁷ SAHLE: *Digitale Editionsformen 2* (wie Anm. 2), 6ff.

⁴⁸ Ebd., S. 14.

schon zu einer „Schärfung und Überprüfung“⁴⁹ der eigenen editionswissenschaftlichen Methoden. Insgesamt blieb der Einfluss all dieser Systeme auf die technologische Entwicklung der Digitalen Edition aber relativ gering, auch wenn natürlich erwähnt werden muss, dass einige dieser Systeme sich nach wie vor großer Beliebtheit erfreuen und in unterschiedlichen Editionsprojekten eingesetzt werden, wie z.B. CTE oder Tustep.⁵⁰

1.2.2 Datenmodellierung

Parallel zu den Entwicklungen des Computers als Hilfsmittel zur Vorbereitung für den Druck und methodisch von grundlegenderer Natur begannen schon früh⁵¹ erste Debatten über geeignete Datenmodelle für geisteswissenschaftliche textbasierte Informationen geführt zu werden. Zu Beginn wurden die Diskussionen besonders auf der Ebene der Datenbankmodellierung geführt. Dabei spielten vor allem zwei konkurrierende Datenbankmodelle eine wichtige Rolle. Zum einen das sich seit den 80er Jahren mehr und mehr durchsetzende relationale Modell und als Alternativansatz dazu das hierarchische.⁵² Die Vorteile des relationalen Modells lagen insbesondere in der sehr schnellen Verarbeitung großer eher flach strukturierter Datenmengen, die sich leicht durch verknüpfte Tabellenstrukturen beschreiben ließen, also hervorragend geeignet waren für die Verwaltung etwa von Metadaten oder Bilddatenbanken. Die Beschränkungen des relationalen Modells traten allerdings dann deutlich zutage, als der Wunsch größer wurde, nicht mehr nur flach strukturierte Daten zu verwalten, sondern die für geisteswissenschaftliche Texte charakteristischen Tiefenstrukturen zu verarbeiten. Gerade im Bereich der dokumentnahen Quellenerschließung waren relationale Datenstrukturen ungeeignet, die für die genaue Texttranskription benötigte Erschließungstiefe zu modellieren. Das relationale Modell hatte seine Vorteile bei der Verwaltung vordefinierter flacher Texteinheiten, die sich auf stark homogene Datenstrukturen anwenden ließen. Diese Daten unterlagen in den meisten Fällen keiner verbindlichen Vorgabe bezüglich der Aufeinanderfolge von Texteinheiten. So ergibt sich der Sinn aus einer Metadaten-Tabelle etwa auch dann noch, wenn die einzelnen

⁴⁹ Ebd., S. 14.

⁵⁰ Bei Tustep liegt dies im wesentlichen daran, dass es inzwischen auf die Entwicklungen im Bereich XML-basierter Markup-Sprachen und Auszeichnungsstandards mit zunehmender Offenheit reagiert.

⁵¹ Spätestens seit den 70er Jahren besonders durch die theoretischen Überlegungen von Dino Buzzetti und von den praktischen Ansätzen von Manfred Thaller beeinflusst.

⁵² Zur Diskussion des Einsatzes von Datenbanken in den Geisteswissenschaften siehe: Manfred THALLER: Data Bases v. Critical Editions, in: Historical Social Research (13.3) 1988, S. 129–139, DERS.: Datenbasen als Editionsformen?, in: Historische Edition und Computer. Möglichkeiten und Probleme interdisziplinärer Textverarbeitung und Textbearbeitung 1989, hrsg. v. Anton SCHWOB/Karin KRANICH-HOFBAUER/Diethard SUNTINGER, S. 215–241, Dino BUZZETTI/Andrea TABARRONI: Informatica e critica del testo: il caso di una tradizione 'fluida', in: Schede umanistiche, N.S., I 2 1991, S. 185–193, Dino BUZZETTI: Digital Editions: Variant Readings and Interpretations, in: ALLC-ACH 96 Conference Abstracts 1996, S. 254–256, URL: <http://gandalf.aksis.uib.no/allc-ach96/Panels/Thaller/buzzetti.html>, Dino BUZZETTI/Andrea TABARRONI: Database Edition of Non-collatable Textual Traditions. In: The Electric Scriptorium: Electronic Approaches to the Imaging, Transcription, Editing, and Analysis of Medieval Manuscript Texts 1995.

Metadaten-Felder in anderer Reihenfolge angeordnet werden. Die semantische Beschreibung des Inhaltes einer Dateneinheit wird dabei meist durch den zugehörigen Feldnamen bestimmt. Bei dokumentenzentrierten (semistrukturierten und unstrukturierten) Daten ist diese Reihenfolge aber gerade nicht beliebig vertauschbar. Um die in historischen Überlieferungen impliziten Informationen explizit formalisieren und speichern zu können, hätte der Dokumenttext anhand der einzelnen Informationseinheiten in seine Einzelteile zerlegt und jeweils in einer Kombination von beschreibender und zu beschreibender Informationseinheit verwaltet werden müssen (z.B. Feldname plus Feldinhalt). Zusätzlich hätte für eine strikte Reihenfolge dieser Informationstupel gesorgt werden müssen, um die richtige Rekonstruktion des Überlieferungstextes gewährleisten zu können. Ein weiteres Problem bei der Verwendung relationaler Modelle bestand darin, dass das Modell vor der eigentlichen Eingabe der Daten schon definiert werden musste bzw. sich nur unter großem Aufwand während der Datenaufnahme modifizieren ließ. Es konnte also nicht flexibel genug auf strukturelle Veränderungen reagieren. Dies ist aber gerade bei der Informationsmodellierung von historischen Überlieferungen unerlässlich, da häufig erst in einem späteren Arbeitsprozess zusätzliche Informationen der Textstruktur zutage treten, die Veränderungen des Modells bewirken. Ändert man das Datenmodell einer Tabelle, betrifft dies immer alle Datensätze dieser Tabelle. Da aber in der Regel keine zwei historischen Dokumente eine exakt identische Struktur besitzen, müsste für jedes Dokument ein eigenes Datenmodell entwickelt werden. Dies ist im Prinzip möglich, führt aber dazu, dass das System z.B. seine Vorteile in Bezug auf die Verarbeitung großer homogener Datenmengen verliert. Ein ebenso großes Problem bestand darin, die unterschiedlichen Informationsschichten, die hierarchisch tief verschachtelt vorliegen konnten, angemessen zu beschreiben. Einmal definierte Datenfelder des relationalen Modells sind atomar in dem Sinne, dass sie keine weiteren Felder anderen oder des gleichen Typs beinhalten können. Kurzum, das Modell war nicht in der Lage, den komplexen, meist tief und heterogen strukturierten Informationsgehalt geisteswissenschaftlicher Textinformationen adäquat modellieren zu können. Einer der ersten, der die Notwendigkeit für ein auf anderen Prinzipien beruhendes Datenmodell erkannte, war Manfred Thaller. Auf Grundlage eines hierarchischen Datenmodells entwickelte er in den 1970er Jahren das Datenbankmanagementsystem Kleio. Das System war in der Lage beliebige Textstrings mit beliebigen hierarchischen Dimensionen von Informationen oder Verarbeitungsanweisungen zu belegen. Das neue an diesem System war die Herangehensweise, nicht nur die Daten in diesem System zu speichern, sondern darüber hinaus auch das Expertenwissen über diese Daten und die Regeln zu ihrer Auswertung.⁵³ Ein großer Nachteil der Datenmodellierung innerhalb von Datenbanksystemen allgemein war die enge Verknüpfung der Modelle mit dem jeweiligen Softwaresystem. Sie bestanden zumeist aus proprietären Formaten, die plattformabhängig für jedes System eigens implementiert werden mussten. Daraus entstand der Wunsch nach einem plattformunabhängigen offenen Modell, das alle Anforderungen in Bezug auf Flexibilität

⁵³ Manfred THALLER: *Texts, Databases, Kleio: A Note on the Architecture of Computer Systems for the Humanities*, in: *Augmenting Comprehension: Digital Tools and the History of Ideas 2004*, hrsg. v. D. BUZZETTI/G. PANCALDI/Short H., S. 49–76.

und Modellierung komplexer hierarchischer Informationsstrukturen erfüllen konnte. Ein Lösungsversuch bestand schließlich darin, das Modell von der konkreten Anwendung zu abstrahieren und in Form von deskriptivem Markup direkt in den Text zu integrieren.

Ausgehend von den ersten Vorstufen von textinternem Markup, das schon in Systemen wie Tustep als prozedurales Markup in Form von Steuerzeichen für Verarbeitungs- oder Formatierungsbefehle eingesetzt wurde,⁵⁴ entwickelte sich mit SGML die Auszeichnungssprache, die wohl bis heute den größten Einfluss auf alle weiteren Entwicklungen im Bereich der deskriptiven, generischen Markup-Sprachen ausübte. Alle wesentlichen Konzepte heutiger Auszeichnungssprachen basieren auf ihren Grundlagen: die Deklaration von deskriptiven Elementen und ihren zugehörigen Attributen, die typische Notation von Elementbeginn und Elementende mit öffnenden und schließenden Klammern, um den Teil des Textes zu markieren, der beschrieben werden soll und ihn gleichzeitig leicht von den beschreibenden Elementen trennen zu können, und die baumartige hierarchische Verschachtelungsmöglichkeit von Elementen ineinander. Dadurch, dass die Modellierung impliziter Informationen durch explizite Beschreibung in Form von Elementen und Attributen jetzt auf der selben Ebene der Ascii-Kodierung stattfand, waren sowohl Daten als auch Datenmodell zwischen unterschiedlichen Systemen plattformunabhängig austauschbar. Die Verschachtelungsmöglichkeiten erlaubten es im Prinzip, beliebig viele Informationsschichten zu kodieren. Durch die Diskussionen um Hypertextstrukturen befeuert und durch die Etablierung des WWW als zentralem Publikationsmedium stark beeinflusst, entwickelte sich aus SGML die Seitenbeschreibungssprache HTML, die im Gegensatz zu SGML aus einem überschaubareren definierten Set an Elementen und Attributen bestand und somit von unterschiedlichen Browserherstellern leichter implementiert werden konnte. Durch die kontinuierlich sinkenden Preise im Bereich der Personal-Computer, das Client-Server-Konzept WWW-basierter Anwendungen und die Möglichkeit, den schon seit langem kritisierten linearen Textfluss mit Hilfe von Hyperlink-Konzepten zu durchbrechen, erschien das WWW als das Medium mit dem größten Potential, die lange schon gestellten Forderungen an eine leichte Zugänglichkeit und eine normierte Datenmodellierung einzulösen. In Folge dieser Entwicklungen entschlossen sich nun auch immer mehr Editionsprojekte, ihre Daten direkt für die Präsentation in HTML zu kodieren oder zumindest in HTML ein alternatives Ziel zur klassischen Druckedition zu begreifen und ihre Projekt auch über WWW-basierte Publikationskanäle zu veröffentlichen.

Allerdings traten auch hier relativ schnell die Nachteile zutage, die mit der Kodierung in HTML verbunden waren. Das eng begrenzte Set an Auszeichnungselementen erfüllte nicht die Anforderungen an eine flexibel erweiterbare und ausdrucksmächtige Beschreibungssprache und führte zudem zu einer ungewollten Vermischung von Informationen der Inhalts- und Ausdrucksebene. Es war vor allem orientiert an layoutbeschreibenden Konzepten wie der Einteilung in Kapitel und Überschriften oder der Formatierung von Textelementen mit Schrifttypen, -größen oder dem Setzen von kursivem oder unterstrichenem Text mittels eigens dafür definierter Elemente. Ein

⁵⁴ Patrick SAHLE: Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels, Bd. 3 Textbegriffe und Recodierung, Norderstedt 2013, 133ff.

erster Lösungsansatz, um die Layoutinformationen vom Markup zu trennen, bestand darin, durch die Verwendung von Stylesheet-Sprachen wie css, diese Informationen separat zu verwalten und mit Klassifizierungen etwa effizient auf definierte Gruppen von Elementen anwenden zu können. Dies verlagerte zwar die typographischen Ausdrucksmittel in einen separaten Bereich, löste aber dennoch nicht das Problem, die Inhaltsebene semantisch beschreiben zu können.

Um die Nachteile des komplexen SGML einerseits und die Beschränkungen von HTML in Bezug auf Ausdrucksmächtigkeit und mangelnde Flexibilität andererseits zu beheben, entwickelte sich schließlich Ende der 90er Jahre die eXtensible Markup Language (XML) mit der Möglichkeit über DTDs oder später auch alternative Schemakonzepte, eigene anwendungsspezifische Beschreibungssprachen definieren zu können.⁵⁵ Einer der wesentlichen Vorteile von XML bestand darin, Datenebene und Präsentationsebene voneinander trennen zu können. So lassen sich formatunabhängige Datenmodelle entwickeln, die die Informationen nicht schon im Hinblick auf ein spezifisches Darstellungsmedium kodieren, sondern davon unabhängig ihre abstrahierte logische Struktur modellieren. Die Trennung von Inhalt und Form erlaubt eine von der Präsentationsform unabhängige Datenhaltung.⁵⁶ Zahlreiche Entwicklungen im Bereich der XML-Technologien bieten darüber hinaus eine breite Palette an Verarbeitungs- und Analysewerkzeugen. XML fungiert somit als medienneutrales (oder auch transmediales⁵⁷), plattform- und anwendungsunabhängiges Datenmodell. Implizite Informationen lassen sich explizit auf der selben technologischen Grundlage (Ascii) beschreiben. Sie sind somit zugleich maschinen- als auch menschenlesbar. Eine „Befundcodierung auf einer dokumentnahen Ebene“⁵⁸ wird ebenso unterstützt wie die Kodierung logischer und funktionaler Eigenschaften. Neben der dokumentzentrierten Kodierung lassen sich ebenso datenzentrierte Kodierungen etwa auf Metainformationsebene realisieren. XML ist software- und plattformunabhängig und lässt sich als ideales Austauschformat zwischen unterschiedlichen Systemen und Applikationen verwenden. Aufgrund der baumartigen, hierarchischen Objektstruktur unterstützt es die inhaltliche Tiefenerschließung von Textdokumenten und ist durch seine Erweiterbarkeit flexibel modifizierbar. Ausgehend von diesen Entwicklungen und auch schon seit SGML parallel dazu hat sich in den vergangenen Jahren die Verwendung der Text Encoding Initiative (TEI)⁵⁹ sowohl als Richtlinieninstrument als auch als Kodierungs-

⁵⁵ Stellvertretend für die vielen positiven Einschätzungen, dieses Konzept der erweiterbaren generischen Markup-Sprachen gewinnbringend bei der digitalen Texterschließung einzusetzen vgl. Sahle: „Mit generischem Markup können Beschreibungssprachen entwickelt werden, die eine an Strukturen, visuellen (kodikologischen, bibliographischen, paläographischen, typografischen) Befunden und inhaltlichen Deutungen orientierte Verzeichnung erlauben, die sehr viel genauer und ausdrucksmächtiger ist, als dies in anderen Technologien und Arbeitsweisen möglich wäre“ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 75.

⁵⁶ „Erst die Auszeichnungssprachen erlauben die Entwicklung informationsreicher digitaler Editionen, deren Datenschicht von ihrer Publikationsschicht vollkommen getrennt ist.“ ebd., S. 162.

⁵⁷ Ebd., S. 164.

⁵⁸ Ebd., S. 75.

⁵⁹ Vgl. C.M. SPERBERG-MCQUEEN/Lou BURNARD/Syd BAUMAN: TEI P5: Guidelines for Electronic Text Encoding and Interchange, Charlottesville 2013, URL: <http://www.tei-c.org/Vault/P5/2.0.1/doc/tei-p5-doc/en/Guidelines.pdf>.

standard etabliert.⁶⁰ Die ersten Versionen (P1-P3) wurden in SGML implementiert, seit der Version P4 (2002) basieren sie auf XML. Die TEI ist eine anwendungsspezifische Beschreibungssprache, die konkrete Textmodelle zur Kodierung formaler, struktureller und inhaltlicher Merkmale unterschiedlicher Textgattungen zur Verfügung stellt. Sie erhebt dabei den Anspruch, disziplinenübergreifend und umfassend zu sein: „support the encoding of all kinds of features of all kinds of texts studied by researchers.“⁶¹ Historisch bedingt unterstützt sie dabei vor allem linguistische und literaturwissenschaftliche Anwendungsszenarien, hat sich darüber hinaus aber auch in anderen Fachdisziplinen durchsetzen können. Es existiert heute eine große Zahl an Initiativen, die innerhalb der TEI spezielle Untergruppen bilden oder auch parallele Projekte entwickeln, die bestrebt sind, Bestandteil der TEI zu werden, wie etwa die CEI (Charters Encoding Initiative),⁶² die sich mit der Beschreibung mittelalterlicher Urkunden befasst, EpiDoc⁶³ im Bereich der Epigraphik und Papyrologie oder auch die Music Encoding Initiative (MEI)⁶⁴. In den meisten digitalen Editionsprojekten wird heutzutage XML zur Informationskodierung verwendet, sei es in Form eines Formats, wie eines der gerade erwähnten oder in Form von spezifischen Eigenentwicklungen, die exakt auf die Anforderungen des jeweiligen Projekts zugeschnitten sind.

Die soeben in kurzer Form beschriebenen Entwicklungen im Bereich der Datenmodellierung hin zu hierarchischen, Markup-basierten Datenmodellen werden bis heute auch durchaus kritisch diskutiert. Im folgenden sollen die Vor- und Nachteile dieser Entwicklungen näher beleuchtet und im Hinblick auf ihre Brauchbarkeit für die in dieser Arbeit angestrebten Modellierungsmöglichkeiten textgenetischer Prozesse betrachtet werden. Zuerst einmal ist zu fragen, ob es sich bei der Trennung von Inhalt und Form auch wirklich um eine solche handelt und wenn ja, ob diese auch in jedem Fall so erwünscht ist.⁶⁵ Wenn von den Vorteilen der Trennung zwischen Inhalt und Form in deskriptiven Markup-Sprachen die Rede ist, so kann sich dies auf unterschiedliche Ebenen beziehen.⁶⁶ Zum einen kann damit die problematische Vermi-

⁶⁰ Obwohl die TEI immer wieder als Standard bezeichnet wird („provide a standard format for data interchange“ [<http://www.tei-c.org/release/doc/tei-p5-doc/en/html/AB.html>]), ist diese Sichtweise nicht unumstritten und soll später noch näher diskutiert werden.

⁶¹ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/AB.html#ABTEI2>

⁶² <http://www.cei.lmu.de/>

⁶³ <http://epidoc.sourceforge.net/resources.shtml>

⁶⁴ <http://music-encoding.org/activities/events>

⁶⁵ Sahle gibt zu bedenken, dass der „Inhalt“ nicht vollkommen unabhängig von der Form existieren kann: „Der Inhalt ist zunächst offensichtlich abhängig von der Form der überlieferten Dokumente, die recodierend aufgenommen und interpretiert wird. Er ist dann grundsätzlich unabhängig von der Form der Präsentation in einer publizierten Edition. Streng genommen kann er aber nicht vollständig unabhängig sein von der schließlichen Form der Veröffentlichung. Der digital codierte Inhalt ist zwar nicht materiell, sondern logisch definiert, trotzdem ist diese Logik nicht unabhängig von jenen medialen Horizonten, die implizit immer mit gedacht werden und denen wir uns nicht entziehen können.“ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 163.

⁶⁶ Dies zumindest, wenn man als Form nicht nur formatierungstechnische Aspekte betrachtet, die ein spezifisches Layout des Textes beschreiben, sondern alle Aussagen, die über diesen Text gemacht werden können, aber selbst nicht Bestandteil des Textes sind, seien sie struktureller, logischer, funktionaler oder semantischer Natur. Wobei es auch schon bei Layoutinformationen, wie dem Einrücken eines Zitates oder Hervorheben bestimmter Textstellen durch Kursivsetzung oder Unterstreichung zu unterschiedlichen Ansichten darüber kommen kann, ob nicht gerade

schung gemeint sein, wie sie früher häufig in HTML-kodierten Texten zu finden war, wenn Überschriften durch Elemente ausgezeichnet wurden, die in der Ausgabe von HTML-Browsern mit einer festen Darstellungsform verbunden waren, wie bei `<h1>`, `<h2>` etc. (hier in der Ausgabe meist durch fixe Textgröße und Zeilenumbruch realisiert). Dieses Problem lässt sich mit Hilfe des deskriptiven Markups leicht lösen, indem z.B. ein ausgabeneutrales Element (`<head>`) definiert wird und durch entsprechende Transformationen und Stylesheets zu einem späteren Zeitpunkt dann die Formatierungseigenschaften für jede beliebige Ausgabeform separat produziert werden.

Es gibt aber noch weitere Ebenen, die vom jeweiligen Textverständnis oder besser dem zugrundeliegenden Textbegriff abhängen. Die Trennung etwa nach dem Prinzip 'alles, was nicht Markup ist, ist Text (Inhalt)' ist nicht unproblematisch. Die Kodierung von alternativen Lesarten ist mit diesem Prinzip nicht zufriedenstellend lösbar, zumindest dann nicht, wenn die Wahrscheinlichkeit für alle Lesarten identisch ist. Geht man von dem einfachen Fall von zwei unterschiedlichen Lesarten aus, so können diese entweder beide im Text kodiert werden `<lesartA> ... </lesartA>` `<lesartB> ... </lesartB>` oder als Attribut im Markup `<lesartA value="...">` `<lesartB value="...">` oder es könnte eine Lesart bevorzugt und im Text kodiert werden, während die zweite ins Markup verbannt wird: `<lesartA value="..."> B </lesartA>` (oder umgekehrt). Im ersten Fall wären beide Lesarten konstitutiv für den Text, was aber nicht der Realität entspricht, da der Autor nur eine von beiden gemeint haben kann. Im zweiten wären beide nicht Bestandteil des Textes, was natürlich auch nicht stimmen kann und im dritten ist es nicht entscheidbar, da die vom Editor bevorzugte Lesart entweder falsch oder richtig sein kann. Wendet man einen dynamischen Textbegriff an, im Sinne der Textgenese z.B., so ist der Inhalt des Textes davon abhängig, zu welchem Zeitpunkt man ihn betrachtet. Eine Hinzufügung einer Textpassage ist erst ab dem Zeitpunkt des Einfügens Bestandteil des Inhaltes, davor war sie es nicht. Andersherum ist eine Streichung ab diesem Zeitpunkt dann nicht mehr Teil des Textes (oder doch? auf der rein physischen Ebene schon, auf einer logischen Ebene nicht). Um eine klare Trennung von Textinhalt und Aussage über diesen Inhalt mit der 'alles, was nicht Markup ist, ist Text'-Methode zu erreichen, müsste bei jeder Modifikation des Textes in Form einer Streichung dieser Streichungstext in das Markup geschrieben und aus dem Textinhalt gelöscht werden. Auch in der Editionspraxis verläuft die Trennung von Inhalt und Aussage über den Text meist nicht so stringent. Es ist z.B. gängige Praxis, Metadaten innerhalb des selben Markup-Dokumentes zu kodieren, das auch den Überlieferungstext beinhaltet, und zwar mit den selben strukturellen Mitteln, als Inhalte von Markup-Elementen. Die eigentliche Trennung wird dann meist durch unterschiedliche Separierungsstrukturen erreicht, die die Metadaten in `<header>` und den Dokumentkörper in `<body>` kapseln, so wie es von der TEI empfohlen wird.

Ein anderer Kritikpunkt betrifft die hohe Komplexität in Bezug auf die zu kodierenden Informationsebenen und den damit verbundenen hohen Arbeitsaufwand.⁶⁷

diese Textmerkmale konstitutiv für die Identität des Textes sind. Aus einer stark dokumentnahen Sicht würde man wohl dazu tendieren. Vgl. hierzu die unterschiedlichen Textbegriffe bei Sahle SAHLE: Digitale Editionsformen 3 (wie Anm. 54), 244ff.

⁶⁷ „Auf der anderen Seite bedeutet diese 'Einfachheit' nicht unbedingt eine Vereinfachung der editorischen Arbeiten. Mit dem Übergang zu insbesondere SGML, dann auch noch zu XML

Schon Renear spricht von der „complexity that taxes the limitations of markup.“⁶⁸ Zum selben Schluss gelangen auch Schmidt, Brocca und Fiormonte: „It is now generally recognised that written texts can contain complexities and subtleties of structure that defeat the power of markup alone to represent them.“⁶⁹ Die Komplexität ergibt sich aus den unterschiedlichsten Auszeichnungsebenen, die alle auf eine Überlieferung in Form von dokumentnahem, topographischem, strukturellem, logischem, analytischem oder funktionalem Markup angewendet werden können. Besonders in genetischen Editionen kann dies zu hochkomplexen Überschneidungen unterschiedlicher Ebenen führen: eine Ebene könnte z.B. die Binnenvarianz von Einfügungs-, Streichungs oder Umstellungsoperationen beschreiben, eine zweite kodiert dazu die in externen Textzeugen vorhandenen Varianten, eine dritte enthält topographische Informationen, um diplomatische Darstellungen zu ermöglichen und eine vierte verzeichnet kritische Textanmerkungen. In offenen kollaborativen Arbeitsumgebungen entsteht ein zusätzliches Problem. Die Auszeichnung des Textes stellt hier nur eine mögliche Interpretation eines Bearbeiters dar. Eine andere Interpretation erfordert wiederum das Einfügen einer neuen Interpretationsschicht. Einer der Vorteile Digitaler Editionen besteht ja gerade in der Möglichkeit, eine Edition als fortlaufenden Prozess zu betrachten und die transkribierten Überlieferungen immer wieder mit neuen Informationsebenen anzureichern, so wie es das Pyramidenmodell von Jenks vorsieht, und dies nicht nur für eine wissenschaftliche Disziplin, sondern am besten für alle, die ihre ganz eigenen fachspezifischen Fragestellungen an den Text haben. Linguistische Analysen etwa bedürfen ganz anderer Markup-Ebenen als historische. In Form von inline-Markup zumindest, stößt man hier schnell an Grenzen. Zum einen an ganz praktische Grenzen, die die oft zitierte Lesbarkeit und Bearbeitbarkeit betreffen,⁷⁰ zum anderen an technische Grenzen, die mit dem Datenmodell zusammenhängen auf dem SGML und XML basieren. Das Modell von SGML oder XML besteht aus einer Baumstruktur, die zwar ineinander verschachtelte Hierarchien erlaubt, aber nur solche, die sich nicht überlappen. Da sich aber unterschiedliche Informationsebenen sehr häufig überlappend überschneiden, müssen unterschiedliche Hilfskonstruktionen

war vielmehr eine Zunahme an Komplexität und Arbeitsaufwand verbunden.“ DERS.: Digitale Editionsformen 2 (wie Anm. 2), S. 73.

⁶⁸ Allen RENEAR: Out of Praxis: Three (Meta) Theories of Textuality, in: Electronic Text 1997, hrsg. v. K. SUTHERLAND, S. 107–126, hier S. 121.

⁶⁹ Desmond SCHMIDT/Nicoletta BROCCA/Domenico FIORMONTE: A Multi-version Wiki, in: Digital Humanities 08 2008, S. 187–188, URL: www.ekl.oulu.fi/dh2008/Digital%20Humanities%202008%20Book%20of%20Abstracts.pdf, hier S. 187.

⁷⁰ Die vielzitierte Lesbarkeit ist einem solchen Szenario eher theoretischer Natur. Da sich sowohl Markup, als auch Text im selben Ascii-Zeichenraum befinden spricht man häufig davon, dass sie menschenlesbar sind und man lediglich einen einfachen Ascii-Editor dazu benötige. Dies trifft aber nur auf relativ schwach ausgezeichnete Texte zu. Komplexe Auszeichnungsebenen, wie die hier beschriebenen, verlangen nach spezifischen Softwaresystemen, um dem Benutzer jeweils nur die Information anzuzeigen, die er für seine Aufgabe benötigt und alle übrigen Ebenen vor ihm zu verbergen. Auf der anderen Seite erscheint das Argument an sich fragwürdig. Der eigentliche Mehrwert einer Informationskodierenden Technologie kann nicht darin bestehen, dass sie vom Menschen lesbar ist, sondern, dass sie mit dem geringst möglichen Informationsverlust erkenntnisrelevante Information in maschinensprachliche Information übersetzt. Die Schicht die für diese Übersetzung verantwortlich ist, sollte unabhängig von ihrer menschlichen Lesbarkeit bewertet werden.

dafür sorgen, dass diese Überlappungen kodiert werden können und gleichzeitig regelkonformes Markup erzeugt wird. Es gibt verschiedene Techniken, um dies zu erreichen. Innerhalb der deskriptiven Markup-Sprachen verwendet man in der Regel Fragmentierungs-, Milestone- oder auch stand-off-Markup-Lösungen,⁷¹ die alle wieder ihre Nachteile mit sich bringen. Fragmentierungen weisen den Nachteil auf, dass logisch zusammenhängende Informationseinheiten getrennt werden und durch Referenzsysteme ihre Zusammengehörigkeit extra kodiert werden muss. Die inhaltsleeren Milestone-Lösungen widersprechen der eigentlichen Idee des deskriptiven Markups, Textinhalte in Elementen zu kapseln und somit einen nichtleeren Inhalt zu beschreiben. Darüber hinaus laufen sie dem Grundgedanken des hierarchischen Textmodells zuwider, indem sie nur noch als Ankerpunkte dienen (mit jeweiliger Referenz auf den start- oder/und end-Milestone). Dies erschwert die Verarbeitung durch verwandte Technologien wie XPath, XQuery oder XSLT erheblich, da diese Technologien selbst auf diesem Prinzip aufgebaut sind und sich Abfragen und Adressierungen in der Regel auf Elementinhalte beziehen. Darüber hinaus lassen sich mit Milestones keine komplexen Elemente anlegen, somit kann z.B. innerhalb eines Schemas nicht der Inhalt zwischen zwei Milestones definiert werden, was wiederum die Validierung der Datenstruktur erschwert. Stand-off-Lösungen basieren auf der Idee, unterschiedliche Hierarchien ausserhalb des eigentlichen Dokumentkörpers zu kodieren. Dies kann innerhalb derselben Datei oder in externen Dateien geschehen. Die ausserhalb kodierten Markup-Strukturen verweisen dann mit Hilfe von Identifikatoren auf einzelne Elemente oder Gruppen von Elementen oder direkt auf die Anfangs- und Endpositionen des plain-Textes. Bei letzterer Methode besteht der Nachteil darin, dass bei textuellen Veränderungen am Zieldokument gleichzeitig immer auch die Char-Positionen des stand-off-Markups aktualisiert werden müssen.⁷² Bei den vorgestellten Lösungsansätzen des Fragmentierens und der Verwendung von Milestones besteht ein weiterer Nachteil darin, dass eine Informationsebene immer den Vorzug vor den übrigen erhält.⁷³ Alle anderen Ebenen, die mit diesen Hilfskonstrukten kodiert werden, haben dann mit den Einschränkungen bezüglich ihrer Verarbeitbarkeit zu leben.⁷⁴ Ausgehend von allgemeinen texttheoretischen Überlegungen, ob hierarchisches Markup überhaupt das geeignete Werkzeug ist, zugleich Struktur (Ausdruck) und Inhalt von

⁷¹ des öfteren auch non-embedded genannt im Gegensatz zu embedded- oder inline-Markup

⁷² <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/NH.html> für eine genaue Beschreibung der Einschränkungen vor allem von stand-off-Methoden der TEI vgl. Piotr BANSKI: Why TEI stand-off annotation doesn't quite work: and why you might want to use it nevertheless, in: Balisage Series on Markup Technologies 5 2010, URL: <http://www.balisage.net/Proceedings/vol5/html/Banski01/BalisageVol5-Banski01.html>.

⁷³ Mit Hilfe der Milestone-Lösung kann ein Dokument im Prinzip zwar auch vollständig hierarchielos kodiert werden, indem alle Ebenen durch Milestones modelliert werden, zu bedenken ist dabei aber, dass es dann überhaupt keine Hierarchien mehr gibt und dies prinzipiell die Verwendung von hierarchischem Markup in Frage stellt.

⁷⁴ Vgl. hierzu „Mehrfache Sichten und primäre Sichten“ SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 361 oder auch Alexander CZMIEL: Adäquate Markupsysteme für die digitale Behandlung altägyptischer Texte (Magisterarbeit), Universität zu Köln 2003, URL: <http://old.hki.uni-koeln.de/studium/MA/>, S. 7.

Texten zu beschreiben, üben vor allem Buzzetti⁷⁵ und Greenstein⁷⁶ Kritik an hierarchischen Markup-Sprachen.⁷⁷ Buzzetti betont dabei vor allem den Zusammenhang zwischen semantischem Inhaltsmodell und syntaktischer Struktur der linearen Textdarstellung. Er unterteilt textuelle Ebenen in Modelle, die die äußere Struktur (die Ausdrucksseite) eines Textes beschreiben und jene, die von diesen unabhängig, die inhaltliche Struktur beschreiben. Da Markup selbst linear im Textfluss verankert sei (linear-hierarchisches Prinzip), gehöre es aber selbst zur Textstruktur. In diesem Sinne verhielte es sich damit nicht anders als andere diakritische Zeichen, wie z.B. Interpunktionszeichen. Markup müsste in seiner linearen Natur in der Lage sein, nicht-lineare (semantische) Strukturen zu beschreiben. Hier sieht Buzzetti das eigentliche Problem. Ein von der Ausdrucksseite unabhängiges Strukturmodell des Inhalts könne eher mit Techniken des non-embedded-Markup oder gleich mit geeigneten Datenbankmodellen beschrieben werden. Dieses abstraktere Modell sei dann losgelöst vom eigentlichen Text und beschreibe nicht wie Markup die formale syntaktische Struktur, sondern die inhaltlich semantische.⁷⁸ Auch für Greenstein stößt das hierarchische Markup-Modell (vor allem in seiner OHCO-These als geordnete Hierarchie⁷⁹) dort an seine Grenzen, wo nicht nur allein das linguistische Konzept, sondern das logische zu modellieren ist, da das Markup in seiner doppelten linearen Natur auch in den geordneten Hierarchien linear ist. Es existierten aber durchaus auch Textmodelle, die sich weder durch linearen Text noch durch geordnete Hierarchien adäquat beschreiben ließen, z.B. Textmodelle als semantische Netzwerke.⁸⁰ Auch aus technischer Sicht wurde immer wieder Kritik an einem allzu ausschließlichen Gebrauch von hierarchischem Markup geübt. So sieht Thaller XML-basierte Datenmodelle zwar als durchaus geeignet an, um als Austauschformat zwischen eigenständigen Software-Applikationen zu fungieren, aber nicht als geeignet, um das interne Datenmodell eines Softwaresystems zu bilden, da in diesem eng verzahnte Softwaremodule einen Datenaustausch innerhalb von Sekundenbruchteilen erforderten. Für geisteswissenschaftliche Daten identifiziert er mindestens 3 unterschiedliche Szenarien, die mit Hilfe unterschiedlicher Werkzeuge der Informationstechnologie gelöst werden können, von denen nur für ein Szenario der Einsatz von Markup sinnvoll sei: 1. Datenbanktechnologien für aussagekräftige Abfragen (um z.B. Relationen von klar getrennten Texteinheiten in Form von Records zu verarbeiten), 2. Textverarbeitungswerkzeuge, die in der Lage sind, Eigenschaften von strukturierten Texten oder Textfragmenten auszuwerten und abzubilden, die nicht sinnvoll auf Datenbankebene zu modellieren sind, da sie nicht auf die atomaren Einheiten eines Datenbankmodells herunter gebrochen werden können (also die Markup-Ebene) und

⁷⁵ Dino BUZZETTI: Diacritical ambiguity and markup, in: *Augmenting Comprehension: Digital Tools and the History of Ideas 2004*, hrsg. v. D. BUZZETTI/G. PANCALDI/Short H., S. 175–188.

⁷⁶ Daniel GREENSTEIN: Conceptual Models and Model Solutions: A Summary Report of the TEI's Working Group on Historical Studies, in: *Modelling Historical Data: Towards a Standard for Encoding and Exchanging Machine-Readable Texts. (A.11) 1991*, S. 195–204.

⁷⁷ Die Kritik bezieht sich vor allem auf das eingebettete Markup.

⁷⁸ BUZZETTI: Diacritical ambiguity (wie Anm. 75), S. 179.

⁷⁹ Allan RENEAR/Elli MYLONAS/David DURAND: Refining our Notion of What Text Really Is: The Problem of Overlapping Hierarchies, in: *Text Encoding Initiative: Background and Contexts 1993*, URL: <http://www.stg.brown.edu/resources/stg/monographs/ohco.html>.

⁸⁰ Vgl. GREENSTEIN: Models (wie Anm. 76), S. 202.

3. die analytische Ebene, wissensbasierte Werkzeuge, die semantische Analysen von Texten oder Textfragmenten zulassen. Darüber hinaus kritisiert er, dass heutzutage die Tendenz dahin geht, alle Textphänomene, die den drei unterschiedlichen Ebenen zugeordnet werden können, mit ein und demselben Werkzeug zu verarbeiten, nämlich XML-basiertem Markup. Er plädiert hier für eine klare Trennung und unterschiedliche Modellierung und Verarbeitung dieser Ebenen.⁸¹ Seiner Meinung nach sollte bei dem aktuellen 'Hype' um Markup-Sprachen auch nicht aus den Augen verloren werden, dass es nicht nur um Werkzeuge gehen kann, die mit Markup versehenen Text verarbeiten, sondern auch um solche, die aus der Analyse von Text die gleiche Funktionalität gewinnen, die die Auszeichnung von Markup ergibt. Somit könnte auch der enorme Aufwand, der betrieben wird, um Texte mit Markup zu versehen, erheblich verringert werden.

Die TEI stellt in diesem Sinne eine von vielen möglichen Lösungen dar, geisteswissenschaftliche Texte zu modellieren. Da sie inzwischen aber faktisch zum meistverwendeten Auszeichnungsparadigma innerhalb der Geisteswissenschaften geworden ist, sollen kurz die wesentlichen Vor- und Nachteile skizziert werden. Wie bereits angedeutet, liegt einer der wesentlichen Vorteile bei der Verwendung der TEI darin, erstmals (zumindest ansatzweise) über ein genormtes Set von Auszeichnungselementen zu verfügen, mit denen Textphänomene auf verschiedenen Interessenebenen unterschiedlicher Fachdisziplinen modelliert werden können. Insofern stellt sie also eine Art Standard dar, der es erlaubt, mit dem gleichen Vokabular unterschiedliche Arten von Texten zu beschreiben, analytischen Prozessen zugänglich zu machen und auf der Softwareebene für eine Austauschbarkeit der Daten mit möglichst geringem Informationsverlust zu sorgen.⁸² Nach dem Selbstverständnis der TEI liegt ihr kein bestimmter Textbegriff zugrunde. Sie erhebt den Anspruch, losgelöst von solchen alle denkbaren Modelle zu unterstützen. Sie versteht sich dabei zugleich auch als offener Standard, der es jeder Disziplin erlaubt, die vorgegebenen Richtlinien um eigene zu erweitern und in den Standard zu integrieren.⁸³ Zum einen bietet sie die Möglichkeit, sich in Form von special interest groups zusammenzuschließen und sich dort mit speziellen Problemen auseinandersetzen, wie z.B. der Kodierung genetischer Text-

⁸¹ THALLER: Kleio (wie Anm. 53), 69ff.

⁸² „Not having to configure software for multiple formats should, *ceteris paribus*, minimize the need for one-off integration and make the development and maintenance of tools and procedures less error-prone“ Thomas HANSEN: TEI - Keeping It Simple, in: Digital Medievalist 7 2011, URL: <http://www.digitalmedievalist.org/journal/7/hansen/>, §9.

⁸³ „In other words, TEI provides a format which on the one hand is general and popular, and, on the other, articulate and flexible.“ebd., §8 oder auch Ritter: „Für die verschiedenen Textgattungen werden angepasste Schemata empfohlen, mit denen die sich aus der Analyse der Textgrundlage ergebenden Strukturen und Besonderheiten durch geeignete XML-Kodierungen abgebildet werden können. Diese Vorgehensweise zeichnet sich durch eine große Flexibilität aus, da auch Sonderfälle durch Hinzunahme von empfohlenen Auszeichnungen anderer Textgattungen abgedeckt werden können. Auch die Einschränkung des Markups einschließlich der Menge der erlaubten Attribute ist möglich, um eine Vereinheitlichung der verwendeten Tags bei verteilter Arbeitsweise zu erreichen“ Jörg RITTER/Susanne SCHÜTZ/Stefan TEITGE: Entwicklung und Einsatz einer TEI-konformen Arbeitsumgebung für die Edition der Dramen von Karl Ferdinand Gutzkow, in: Jahrbuch für Computerphilologie 9 2007, URL: <http://computerphilologie.digital-humanities.de/jg07/risquet.html>, §17.

strukturen. Nach einem Evaluierungsprozess werden dann in der Regel die Ergebnisse in Form von neuen Richtlinien in die Schemadefinitionen der TEI-integriert. Zum anderen besteht aber auch die Möglichkeit, eigene TEI-fremde Elementsetdefinitionen in den TEI-konformen Code zu integrieren. Trotz oder auch gerade wegen dieses Konzeptes mangelt es aber auch nicht an Kritik an den Richtlinien. Zum einen wird der Anspruch, als Grundlage für alle möglichen Textbegriffe zu fungieren kontrovers diskutiert. Durch die enge Bindung an das OHCO-Modell⁸⁴ unterstützt sie eben nur eine bestimmte Sicht auf den Text, die als geordnete Hierarchie. So bevorzugt sie das Textverständnis ganz bestimmter Fachdisziplinen. Darüber hinaus ist ihre Sichtweise sehr auf gedeutete (logische) Strukturen von Textphänomenen bezogen und vernachlässigt dabei die physischen und visuellen Eigenschaften⁸⁵ und präferiert eindeutig die Hierarchie einer Informationsebene vor vielen anderen.⁸⁶ Andererseits ist auch die Bezeichnung Standard nur in einem sehr weitläufigen Sinne zutreffend. Durch die prinzipielle Offenheit für eigene Markup-Strukturen und vor allem durch die Tatsache, dass ein und das selbe Textphänomen mit den unterschiedlichsten Methoden ausgezeichnet werden kann, stellt sich ernsthaft die Frage, worin dann noch dieser Standard, im Sinne einer verbindlichen Norm, besteht. So ist die Variantenverzeichnung mit Hilfe des apparatus-Elements auf viele unterschiedliche Arten möglich, um nur ein Beispiel zu nennen (mehr zu dieser Problematik siehe Abschnitt 2.6). Aus dieser Tatsache folgt auch die bis heute mangelhafte Unterstützung von spezifischen TEI-verarbeitenden Softwaretools. Diese müssen im Prinzip alle unterschiedlichen Auszeichnungstechniken auf der Applikationsebene implementieren. Kommen dann noch zusätzliche externe Markupdefinitionen hinzu, die in die TEI-Strukturen eingebettet sind, muss die Applikation zusätzliche Anweisungen enthalten, die es ihr erlaubt, auch diese Informationen zu verarbeiten. Ein minimales TEI-konformes Dokument muss lediglich aus einem TEI-Header und aus einem nicht weiter definiertem Textteil bestehen. Besteht der Textteil aus komplett TEI-fremdem Code, dann besteht zwar ein rudimentärer Rahmen, die TEI-verarbeitende Software 'weiß' aber nicht, wie sie den fremden Code behandeln soll. Somit handelt es sich um ein Format, das keine großen Vorteile in Bezug auf genormte Verarbeitung, Analyse oder Darstellung des eigentlichen Dokumentinhaltes bietet.⁸⁷

Insgesamt ist dieses Dilemma aber auch nicht zufriedenstellend für alle Ansprüche zu lösen. Der einen Benutzergruppe geht die verbindliche Einigung auf festgelegte Auszeichnungselemente und -strukturen nicht weit genug, da nur bei einer strengen Einhaltung der Richtlinien sinnvolle Softwaretools entwickelt werden können, die in der Lage sind, anhand vorhersagbarer Strukturen analytische Prozesse implementie-

⁸⁴ siehe auch James CUMMINGS: *A Companion to Digital Literary Studies*, (Part IV Methodologies: The Text Encoding Initiative and the Study of Literature), hrsg. v. Susan SCHREIBMAN/Ray SIEMENS, Oxford 2008, URL: <http://www.digitalhumanities.org/companionDLS/>, §25.

⁸⁵ Vgl. SAHLE: *Digitale Editionsformen 3* (wie Anm. 54), S. 365.

⁸⁶ Vgl. ebd., S. 361.

⁸⁷ so kritisiert z.B. Keating, dass die früheren Versprechungen der TEI nach „pan-comparison of documents or cross-project document-encoding exchange“ bis heute nicht eingelöst werden konnten. John G. KEATING u. a.: *A Digital Edition of a Spanish 18th Century Account Book: Part 1 – User Driven Digitisation*, in: *Jahrbuch für Computerphilologie* 5 1010, URL: <http://computerphilologie.digital-humanities.de/jg08/keating1.html>.

ren zu können. Anderen Benutzern gehen diese Vorgaben schon zu weit.⁸⁸ Bei dieser Gruppe bestehen (durchaus berechtigte) Zweifel daran, dass eine zu strenge Standardisierung nicht ihren spezifischen Anforderungen Rechnung tragen würde. Es kommt so z.B. immer wieder zu Situationen, in denen Elemente zweckentfremdet und nicht für das ursprünglich gedachte Konzept verwendet werden.⁸⁹ Diese Problematik schien den Entwicklern der TEI so nicht bewusst gewesen zu sein. Das Textverständnis der TEI geht davon aus, dass sich die meisten Textphänomene durch genormte Auszeichnungselemente beschreiben lassen.⁹⁰ Dass dem nicht zwangsläufig so ist, zeigt z.B. Keating,⁹¹ der gerade in der Festlegung der Bedeutung von Textphänomenen durch die Elementnamen der TEI einen großen Nachteil sieht.⁹² Der Dokumenttext und seine einzelnen Elemente sind nicht losgelöst von ihrem Kontext und ihrer Funktion zu betrachten. Innerhalb verschiedener Kontexte können vermeintlich gleiche Textstrukturen unterschiedliche Bedeutungen haben, die semantisch nicht immer korrekt mit Elementen der TEI beschrieben werden können.⁹³ In eine ähnliche Richtung geht die Kritik von Bart. Ihr fehlt es bei der TEI an experimentellem Charakter. TEI-Markup sei geeignet, um Auszeichnungen von Phänomenen vorzunehmen, die vorher schon bekannt seien. Es fehle aber an Möglichkeiten temporäre Annotationen zu verwenden, die evtl. erst bei einem späteren Analysieren als relevant erkannt werden.⁹⁴

⁸⁸ „We argue that while the standard performs well as a sustainable storage and interchange format, it is generally too complicated to operate efficiently“ HANSEN: Keeping It Simple (wie Anm. 82).

⁸⁹ vgl. dazu Huitfeldt, der als einen der Gründe angibt, warum man sich nicht für die Richtlinien der TEI entschieden hätte, dass man Elemente hätte zweckentfremden müssen, um die gewünschte Funktionalität zu erreichen. Claus HUITFELDT: Philosophy Case Study, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD/Katherine O'BRIEN O'KEEFFE/John UNSWORTH, S. 181–196.

⁹⁰ „That there are no essays in this volume or sections in the guidelines on encoding cookbooks, newspapers, guidebooks, quotation dictionaries, instruction manuals, commonplace books, or mail-order catalogues is not because they have been considered and found wanting; there just hasn't been space or occasion to discuss them specifically. For the most part these particular genres can be readily handled using the TEI's provisions for encoding prose texts, with the addition of some other elements for their distinctive features. If you are a scholarly editor of texts then the TEI is applicable to your texts.“ John LAVAGNINO: When not to use TEI, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD/Katherine O'BRIEN O'KEEFFE/John UNSWORTH, S. 334–338, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/lavagnino.xml?style=printable, hier S. 334.

⁹¹ G. KEATING u. a.: Account Book (wie Anm. 87).

⁹² Im Gegensatz dazu sieht die TEI gerade ihren großen Nutzen: „It is up to the creators of XML vocabularies (such as these Guidelines) to choose intelligible element names and to define their intended use in text markup. That is the chief purpose of documents such as the TEI Guidelines. From the need to choose element names indicative of function comes the technical term for the name of an element type, which is generic identifier, or GI.“<http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SG.html>

⁹³ „the element types have to fit the content; otherwise, tag abuse and communication breakdown might occur.“ HANSEN: Keeping It Simple (wie Anm. 82), §12.

⁹⁴ „TEI and PPEA recommendations tend for very laudable reasons to address the most common instances in which the significance of a text string is already known or at least easily discernible before markup commences.“ Patricia R. Bart BART: Experimental markup in a TEI-conformant setting, in: Digital Medievalist 1 2006, URL: <http://www.digitalmedievalist.org/journal/2.1/bart/>, §37 und weiter: „As will be seen in the remainder of this discussion, a series not only of experimental attributes but also of experimental elements based on the models of the existing

Parallel zu diesem Hauptstrang der Entwicklung von Digitalen Editionen über den Weg von SGML über XML zu TEI-konformen Kodierungen hat es auch immer wieder alternative Modellierungsansätze gegeben. Sei es, indem zwar auf hierarchische Auszeichnungssprachen wie XML zurückgegriffen wurde, aber bewusst nicht auf die Richtlinien der TEI wie z.B. bei dem Alcalá Account Book-Projekt, einer „Digital Edition of a Spanish 18th-Century Account Book“⁹⁵ oder in anderen Projekten, die generell hierarchische Markup-Sprachen als Lösungskonzept verwarfen, wie etwa im Wittgensteinprojekt.⁹⁶ Besonders die eingeschränkten Möglichkeiten, gleichwertige Informantionsebenen (also konkurrierende Hierarchien) ein und desselben Textes modellieren zu können, führte zu alternativen Markup-Entwicklungen. Hier sind vor allem Ansätze, wie die von Wendell Piez entwickelte Markup-Sprache LMNL⁹⁷ oder das von Claus Huitfeldt entwickelte 'Multi-Element Code System' (MECS)⁹⁸ zu nennen, die ausdrücklich dafür entwickelt wurden, überlappende Strukturen zu unterstützen,⁹⁹ oder das damit im engen Zusammenhang stehende Modell azyklisch gerichteter Graphen (Goddag - General Ordered-Descendant Directed Acyclic Graph),¹⁰⁰ eine Datenstruktur, die komplexere Modellierungen zulässt, als einfache hierarchische XML-Baumstrukturen. Weitere Ansätze, die multiple Hierarchien innerhalb von XML-Technologien zu lösen versuchen sind z.B. die multi-colored trees (MCT)¹⁰¹ oder auch MultiX.¹⁰²

TEI elements, yet named in such a way as to make clear their experimental nature, would foster the combination of rigorous encoding and the free play of ideas needed for the exploration of text in relation to codex, and of large groups of encoded codices in relation to one another, enabling the computer-aided study of large samples of evidence of scribal practice in copying, correction, mise-en-page, ordinatio, glossing, and use of materials such as vellum and paper stocks.“ ebd., §40.

⁹⁵ „Teehan and John Keating outline their innovative approach from a software engineering perspective, analyse encoding requirements and production workflow, and justifying their decision to use their own customised XML schema. As a salutary reminder of the limits as well as utility of the TEI Guidelines“ <http://computerphilologie.digital-humanities.de/jg08/preface.html> §12, wobei dieses Editionsprojekt bei weitem nicht das einzige ist, das eine solche Entscheidung getroffen hat.

⁹⁶ Hier aus Gründen der mangelnden Unterstützung multipler Hierarchien durch SGML. Espen S. ORE/P. CRIPPS: The Electronic Publication of Wittgenstein's Nachlaß, in: The Digital Demotic. DRH97 Conference, Digital Resources for the Humanities 1998, hrsg. v. Lou BURNARD/Marilyn DEEGAN/Harold SHORT, S. 111–118, hier S. 113.

⁹⁷ <http://piez.org/wendell/LMNL/lmnl-page.html>, Wendell PIEZ: LMNL in Miniature. An introduction, in: Amsterdam Goddag Workshop 2008, URL: <http://piez.org/wendell/LMNL/Amsterdam2008/presentation-slides.html>.

⁹⁸ Claus HUITFELDT: MECS (1992). Die BEE ist Wittgenstein's Nachlass. The Bergen Electronic Edition, Oxford 1998.

⁹⁹ CZMIEL: Markupsysteme (wie Anm. 74), S. 20.

¹⁰⁰ C.M. SPERBERG-McQUEEN/Claus HUITFELDT: GODDAG. A Data Structure for Overlapping Hierarchies, in: ALLC-ACH99 1999.

¹⁰¹ <http://ebookbrowse.com/jlssw2004-mct-talk-pdf-d292901133>

¹⁰² <http://www.idealliance.org/papers/extreme/proceedings/html/2007/Chatti01/EML2007Chatti01.html>

1.2.3 Publikation

1.2.3.1 Externe Speichermedien

Neben den Entwicklungen hin zur markuporientierten Datenmodellierung veränderten sich auch die medialen Publikationskanäle Digitaler Editionen. Aktuell dominieren WWW-basierte Editionen. Davor waren es vor allem die externen Speichermedien, zu Beginn noch in Form von Disketten, seit Einführung der CD und später der DVD aber in Regel von diesen abgelöst, die es ermöglichten Editionen in größerem Umfang digital zu publizieren. Die 1988 erschienene Edition „Goethes Werke: Hamburger Ausgabe“ umfasste ganze 67 5 1/4 Zoll Disketten und wurde später von einer CD-Ausgabe abgelöst.¹⁰³ Der Gebrauch dieser Editionen war zumeist an spezifische Softwaresysteme wie das Retrievalprogramm WordCruncher oder Visualisierungsprogramme wie Folio Views angewiesen.¹⁰⁴ Zu dieser Zeit waren die digitalen Versionen in der Regel eins zu eins Umsetzungen der gedruckten Editionen oder enthielten Zusatzmaterial, welches für den Druck zu umfangreich war.¹⁰⁵ Dieser heutzutage eher als Sonderform der Edition bekannte doppelte Publikationsweg (im Druck und auf digitalen Speichermedien), die Hybrid-Edition, war aus damaliger Sicht ein logischer Schritt in Richtung sinnvoller computergestützter Ergänzung zur eigentlich unantastbaren Königsdisziplin der Printedition.¹⁰⁶ Dort, wo sie funktionalen Mehrwert versprach, wie Volltextsuchen oder Portabilität, war sie eine willkommene Zugabe zur Druckedition. Der Druck blieb aber weiterhin „das eigentliche, alles leitende Ziel.“¹⁰⁷ Auch von den Verlagen wurden diese Entwicklungen forciert. Aus bereits bestehenden Printpublikationen konnte auf diesem Wege die Edition auf dem gleichen Vermarktungsweg ein zweites Mal vertrieben werden, meist zu stattlichen Summen.¹⁰⁸ Die Sichtwei-

¹⁰³ <http://www.kisc.meiji.ac.jp/~mandel/recherche/goethe-werk.html#electronic-goethe>

¹⁰⁴ Vg. Walter FANTA: Die Klagenfurter Ausgabe Robert Musil. Historisch-kritisches Edieren am Computer, in: Jahrbuch für Computerphilologie 8 2007, URL: <http://computerphilologie.digital-humanities.de/jg06/fanta.html> oder auch Thobias OTT: Datenaufbereitung für elektronische Publikationen, in: Beihefte zu Editio, Bd 12: Computergestützte Text-Editionen 1999, S. 79–85. Dies führte auch zu erheblichen Problemen bei der Langzeitverfügbarkeit und Migration der so erstellten Daten. Vgl. auch Roland S. KAMZELAK: Zur Nachhaltigkeit von elektronischen Texten: XML und TEI, in: Was ist Textkritik? Zur Geschichte und Relevanz eines Zentralbegriffs der Editionswissenschaft 2009, S. 3–19.

¹⁰⁵ Im Falle der Hamburger Goethe Edition bestand die CD-Ausgabe sogar nur aus einer Teilmenge der Druckedition, Sachregister und Kommentare wurden nicht digitalisiert. <http://www.kisc.meiji.ac.jp/~mandel/recherche/goethe-werk.html#electronic-goethe>

¹⁰⁶ Als Beispiel für eine Hybrid-Edition (CD/Druck) siehe z.B. Roland S. KAMZELAK: The Hybrid Edition of Harry Count Kessler's Diary, in: Perspectives of Scholarly Editing / Perspektiven der Textedition. Hg. von H.T.M. van Vliet und Bodo Plachta. 2002, S. 181–190.

¹⁰⁷ Dirk GÖTTSCHE/Monika ALBRECHT: Ingeborg Bachmanns 'Todesarten'-Projekt. Elektronische Vorbereitung und Erschließung der kritischen Edition, in: Historical Social Research / Historische Sozialforschung (21.3) 1996, S. 154–160.

¹⁰⁸ Aus Sicht der Verlage machte der Vertrieb von CDs und DVDs in dieser Anfangsphase durchaus Sinn. Die Vertriebswege blieben die gleichen wie die der Printpublikationen, die Texte waren vielfach schon vorhanden und so liessen sich relativ kostengünstig die schon auf dem klassischen Wege publizierten Editionen erweitert durch einfache Recherchefunktionen erneut vermarkten. Sahle fasst dies so zusammen: „Wenn etablierte wissenschaftliche Verlage die wichtigsten Texte und Autoren des bestehenden Kanons mit neuen Benutzungsmöglichkeiten auf den Markt brachten, so gab es eine ganze Reihe von Bibliotheken, die um eine Anschaffung praktisch nicht

se der 'Buchedition als alles leitendes Ziel' und die nur zögerliche Anerkennung des digitalen Publikationsmediums war vor allem geleitet durch ein großes Misstrauen gegenüber der „Langfristigkeit, Stabilität, Authentizität und Zitierbarkeit“¹⁰⁹ Digitaler Editionen und dem vermeintlich höheren Prestige der Buchkultur. Es gab aber auch schon erste Projekte, die erkannten, dass ihre Materialien zu umfangreich waren, um auf diesem doppelgleisigen Wege publiziert werden zu können und die dafür sorgten, dass der eigene Stellenwert der Digitalen Publikation an Gewicht gewann. Dies waren z.B. „Wittgenstein's Nachlass: The Bergen Electronic Edition“,¹¹⁰ die von Beginn an als Digitale Edition angelegt worden war, die „elektronische historisch-kritische Musil-Gesamtausgabe“¹¹¹ oder das Canterbury-Tales-Projekt.¹¹² Die Form der Hybrid-Edition war ein typisches Merkmal der Anfangszeit Digitaler Editionen. Durch die fortschreitende Etablierung der Digitalen Edition und die Anerkennung dieser als eigenständige Publikationsform änderte sich auch langsam die Einstellung gegenüber der Druckedition als primärem Ziel und führte zu einer Zwischenstufe, in der beiden Formen der gleiche Stellenwert beigemessen wurde: „the electronic and the printed editions are two different types of edition which each fulfill their own function within scholarly research.“¹¹³ Gerade dort, wo derzeit die großen Vorteile der Digitalen Edition zu finden sind, wurden diese noch in der gedruckten Form gesehen: „Durch die Verzweigung zwischen Transkriptionsedition und kritischer Buch-Edition bleibt einerseits der historisch-kritische Anspruch einer Transparenz aller editorischen Entscheidungen erhalten, andererseits kann sich die Buch-Edition ganz auf die Rekonstruktion und Darbietung der Texte in ihrer Genese konzentrieren.“¹¹⁴ Heutzutage ist es wohl unbestritten, dass die Vorteile, insbesondere was den Vergleich von Textvarianten und die Darstellung der Textgenese betrifft, im digitalen Medium liegen. Diese Vorteile der Digitalen Edition erklären auch, warum die Hybrid-Edition als ein vorübergehendes Phänomen zu betrachten ist und es in Zeiten des Internets auch so gut wie keine gleichzeitig gedruckten und im WWW publizierten Editionen gibt.¹¹⁵ Die

umhin kamen.“ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 29 Das Potential dieser neuen Publikationsmöglichkeiten blieb jedoch allzu häufig ungenutzt. Aus Kostengründen wurden die Editionstexte nur selten dem neuen Medium angepasst, „man machte, was technisch leicht zu realisieren war, was konzeptionell von den bestehenden Technologien unterstützt wurde und was sich vermarkten ließ.“ ebd., S. 30.

¹⁰⁹ Ebd., S. 65.

¹¹⁰ <http://129.177.5.31/documentation/BTE.html>

¹¹¹ <http://wwwg.uni-klu.ac.at/musiledition/>

¹¹² Vgl. Peter ROBINSON/Kevin TAYLOR: Publishing an Electronic Textual Edition: The Case of The Wife of Bath's Prologue on CD-ROM, in: Computers and the Humanities (32.4) 1998, S. 271–284.

¹¹³ H. T. M. van VLIET: Electronic Editions. New Solutions for Old Editing Problems or Old Wine in New Bottles?, in: Variants: The Journal of the European Society for Textual Scholarship 1 2002, S. 63–83.

¹¹⁴ GÖTTSCHE/ALBRECHT: Todesarten (wie Anm. 107), S. 156.

¹¹⁵ Sicherlich spielen hier auch andere Faktoren eine Rolle, wie z.B. der, dass es immer schwieriger wird den Institutionen und Geldgebern, die solche Projekte fördern glaubhaft zu vermitteln, dass es sinnvoll ist, beide Publikationskanäle zu nutzen. In der Regel zieht man sich bei Projektanträgen heute auf die Position zurück, dass (bei umfangreichen Textmengen zumindest) in erster Linie die Digitale Online-Edition das primäre Ziel ist und das Datenmaterial nur noch in Auszügen im Druck zu publizieren oder nach dem print-on-demand-Prinzip zur Verfügung zu stellen.

Nachteile der Publikation auf externen Datenträgern wie CD oder DVD liegen auf der Hand. Die Langlebigkeit der Speichermedien ist sehr begrenzt. Genau wie im Druck, lassen sich einmal publizierte Editionen nicht beliebig aktualisieren. Der prozesshafte Charakter einer Edition als im Prinzip offenes unabgeschlossenes Projekt lässt sich auf diesem Wege nicht realisieren. Ein weiterer Nachteil an der Veröffentlichung auf einem externen Speichermedium speziell im Bereich Digitaler Editionen ist der, dass die Software, die zur Publikation der Editionen wurde, meist nicht zu diesem Zweck entwickelt wurde. Häufig wurde Standardsoftware verwendet, die Text-Daten zwar mit komfortablen Retrieval-Funktionen erschließen konnte, aber auf die spezifischen Bedürfnisse der einzelnen Editionen nicht in ausreichendem Maße einging.

1.2.3.2 WWW

Die stetig steigende Zahl von Digitalen Editionen im WWW lässt eindeutig einen Publikationstrend weg von temporären Speichermedien hin zu den Onlinemedien erkennen. Die Publikation von Digitalen Editionen findet heutzutage fast ausschließlich dort statt, wo sich eine dauerhafte Verfügbarkeit, leichtere Aktualisierungsmöglichkeiten und im Hinblick auf die Entwicklungen kollaborativer Arbeitsumgebungen auch ein gemeinsames vernetztes Arbeiten an (verteilten) Datenbeständen leichter realisieren lässt, als in den Vorgängermedien. Einer der Gründe, warum das WWW so einen großen Einfluss auf die Entwicklungen im Bereich der Digitalen Editionen ausübte, ist die räumliche Trennung der Datenhaltung und Präsentation nach dem Vorbild der Client-Server-Architektur. Systeme, deren Funktionalität zuvor auf externen Datenträgern, wie CD oder DVD, einem lokalen Computersystem oder einem räumlich eng begrenzten Netzwerk beschränkt waren, lassen sich im WWW im Prinzip weltweit nutzen. Für die Publikation von Digitalen Editionen bedeutete dies, dass wesentliche Nachteile früherer Publikationswege wegfielen. Die Editionen waren nicht mehr an physische Datenträger gebunden und konnten somit sehr einfach veröffentlicht werden. Die serverseitige Vorhaltung der Daten und die Übertragung dieser erst zum Zeitpunkt des Abrufs über das HTTP-Protokoll machte eine Duplizierung der Daten auf Trägermedien überflüssig und eröffnete so die Möglichkeit, zu jeder Zeit dem clientseitigen Benutzer den aktuellsten Datenbestand zugänglich zu machen. Somit unterstützte dieses System auf sehr einfache und effiziente Weise die Sicht auf die Edition als einen beliebig fortführbaren und aktualisierbaren Prozess, der etappenweises Publizieren schon vor Projektende und Aktualisierungen auch noch darüber hinaus ermöglichte. Erst jetzt konnten die Vorteile des elektronischen Textes als distributives Phänomen wie Hayles es beschreibt, zum tragen kommen: „With electronic texts, the data files may be on one server and the machine creating the display may be in another location entirely. This means that electronic text exists as a distributed phenomenon.“¹¹⁶ Infrastrukturell begünstigt wurde die Entwicklung durch Institutionen, die in der Lage waren, preiswerte Serverstrukturen zur Verfügung zu stellen, wie beispielsweise den Universitäten und Forschungseinrichtungen angegliederte Rechenzentren.

¹¹⁶ N. Katherine HAYLES: Translating media - Why we should rethink textuality, in: *Yale Journal of Criticism* (16.2) 2003, S. 263–290, hier S. 274.

Nach dem Client-Server-Prinzip stehen heutzutage hinter fast jeder Digitalen Edition Datenbankmanagementsysteme, die serverseitig für Verwaltung, Analyse und Datenretrieval zuständig sind. Auf der Clientseite werden in der Regel mit Hilfe von Webbrowsern die Editions Inhalte visualisiert. Abgesehen von wenigen fachspezifischen DBMS wie Kleio, kamen in der Anfangszeit des WWW, sofern nicht gänzlich auf ein DBMS verzichtet wurde, vor allem relationale Datenbanken zum Einsatz. Hier in der Regel open-source-basierte Anwendungen wie MySQL oder PostgreSQL.¹¹⁷ Seit geraumer Zeit spielen auch XML-basierte Datenbanken, wie Oracle Berkeley DB XML,¹¹⁸ BaseX¹¹⁹ oder eXist-db¹²⁰ eine immer größere Rolle. Dabei resultieren die Vorteile letzterer vor allem daraus, mit XML kodierte Daten auch direkt in diesem Format verwalten zu können. Umständliche Mappingmechanismen, wie sie bei der Speicherung von XML in relationalen Systemen oft von Nöten sind, fallen damit weg. XQuery als Abfragesprache bietet zudem granularere Retrievalmöglichkeiten als ihr relationales Pendant SQL. Als Nachteile sind beispielsweise die noch nicht ausreichend standardisierte Spezifikation von insert-, update- und delete-Operationen zu nennen, die in der Regel von XML-Datenbanksystemen durch eigene Erweiterungen implementiert werden müssen.¹²¹

Auf der Clientseite bildet bis heute HTML die Grundlage der Darstellung von Webinhalten. Das von allen gängigen Browserherstellern unterstützte Format wurde in Richtung XML zum XHTML-Standard erweitert. Die Vorteile sind bis heute auf Seiten der Verlinkbarkeit quasi unbeschränkter Datenmengen und der Möglichkeit, durch diese Hyperlinkstrukturen das lineare Textverständnis zu durchbrechen, zu finden. Allerdings sind gerade diese technologischen Fortschritte von HTML im Kontext geisteswissenschaftlicher Anwendungen auch immer wieder zum Mittelpunkt kritischer Diskussionen geworden, da HTML durch seinen stark vereinfachenden Charakter im Vergleich zu SGML/XML häufig auch als Rückschritt betrachtet wurde.¹²²

¹¹⁷ Bei letzterer handelt es sich im strengen Sinne um eine objektrelationale Datenbank. Ausgelöst durch Entwicklungen im Bereich der objektorientierten Programmierung entwickelten sich Datenbankmodelle, die vor allem die Vererbungshierarchie von Eigenschaften unterstützten. In der Praxis haben diese rein objektorientierten Modelle aber keine große Verbreitung erlangt. Es kommt aber immer wieder zu Mischformen dieses Modells vor allem mit relationalen Systemen, wie das Beispiel von PostgreSQL verdeutlicht.

¹¹⁸ <http://www.oracle.com/technetwork/products/berkeleydb/index-083851.html>

¹¹⁹ Bei BaseX handelt es sich wieder um eine Mischform, die auch relationale Konzepte in der Datenverwaltung übernimmt. So werden die baumartigen XML-Strukturen „auf relationale Repräsentationen übertragen“ <http://www.steinbeis.de/publikationen/transfermagazin/transfer-032011/datenverarbeitung-mit-ausblick.html>

¹²⁰ <http://exist-db.org/exist/index.xml>

¹²¹ Insgesamt ist in Digitalen Editionsprojekten eine zunehmende Bevorzugung sogenannter NoSQL Datenbanken zu beobachten, hier vor allem die soeben erwähnten Dokumentorientierten Datenbanken.

¹²² „Through the success of the graphical browser, HTML very soon became the standard of hypertext and hyperlinking, although the hypertext capabilities of HTML were in fact an oversimplification of the earlier hypertext systems because it lacked structural markup and the possibility to display alternate views. Further, HTML did not and still does not support neither bidirectional and n-way linking nor the rhetoric classification of categories of links.“ Edward VANHOUTTE: Display or Argument: Markup and Visualisation for Electronic Scholarly Editions, in: Standards und Methoden der Volltextdigitalisierung. Beiträge des Internationalen Kolloquiums an

Diese Nachteile wiegen allerdings weniger schwer, wenn man HTML nur als eine von vielen möglichen Präsentationsformen versteht und die Datenhaltung im Sinne der heutzutage üblichen medienneutralen Datenspeicherung realisiert wird. Vektorbasierte Ausgabeformate wie z.B. SVG¹²³ werden in Zukunft wahrscheinlich einen größeren Stellenwert bei der Datenpräsentation einnehmen. Da SVG auf XML basiert, können SVG-Bäume in jedem gängigen Webbrowser über das Document Object Model verarbeitet und dynamisch auf einzelne Element- und Textknoten zugegriffen werden. Bitmap-Graphiken können als Binärdaten in SVG eingebunden werden. Somit lassen sich SVG-Layer über Bitmap-Digitalisate legen und z.B. unterschiedliche Textschichten hervorheben.¹²⁴ Bislang werden solche Techniken aber so gut wie gar nicht bei der Darstellung Digitaler Editionen eingesetzt, was sicherlich auch damit zusammenhängt, dass die Webbrowser erst seit kurzem diese Technologie nativ unterstützen und der Grad der Umsetzung des vom W3C empfohlenen Vektorgraphikstandards durch die verschiedenen Browserhersteller noch recht unterschiedlich ausfällt. Die vorliegende Arbeit soll unter anderem verdeutlichen, welchen Mehrwert diese Technologie für die Analyse und Darstellung Digitaler Editionen bieten kann.

Zu Beginn des WWW dominierten Digitale Editionen, die sich noch stark den klassischen Ausdrucksformen der Printpublikationen verpflichtet fühlten, das Erscheinungsbild. Sie simulierten vor allem deren ordnenden und beschreibenden Layoutstrukturen in Form der seitenweisen Darstellung und des jeweiligen Apparatsystems. Erst langsam (und bis heute auch noch nicht vollständig) setzte sich die Einsicht durch, dass das Potential Digitaler Editionen zu innovativeren Darstellungsformen von Editions Inhalten führen konnte, als die einfache Abbildung des linearen Textes in digitaler Form, erweitert durch mausklicksensitive Verlinkungen zwischen kritischem Apparat und Editionstext. In der Regel bestehen Digitale Editionen heutzutage mindestens aus dem transkribierten Editionstext, den Metadaten, kritischen Anmerkungen in den unterschiedlichsten Darstellungsformen und diversen Retrieval- und Browsingfunktionen. Der im DBMS gespeicherte Editionstext und die Metadaten werden dabei in der Regel mittels serverseitigen Transformationsprozessen wie XSLT in (X)HTML-konformen Output umgewandelt. Je nach dem Grad der Erschließung lassen sich Zusatzinformationen wie textkritische Anmerkungen, Texterläuterungen, Übersetzungen, Lesetexte, diplomatische Umschriften, Abkürzungsaufösungen, (orthographische) Normalisierungen, Lesevarianten, Literaturverweise etc. über klassische Fußnoten unter dem Text, Pop-upfenster oder Tooltips im Text selbst oder in extra ausgewiesenen Bereichen abseits des Textes anzeigen. Bei ausführlicheren Zusatzmaterialien werden diese auch als gesonderter Seiteninhalt präsentiert (z.B. bei längeren Sach- und Personenregistereinträgen, Paratexten und anderen Kontextmaterialien). Neben den Haupttexten können somit auch Zusatzmaterialien einen eigenständigen Stellenwert erlangen, etwa in Form von eigenständigen Bibliographiedatenbanken oder Personenrepositorien, die dann mit ähnlichen oder auch spezifi-

der Universität Trier 2003, hrsg. v. Thomas BURCH u. a., S. 71–96, hier S. 79.

¹²³ <http://www.w3.org/Graphics/SVG/>

¹²⁴ Bisher war dies in HTML nur durch die sehr rudimentäre Funktionalität der Verweis-sensitiven Graphiken (Image Maps) möglich.

scheren Retrieval- und Browsingfunktionen konsultiert werden können. So lassen sich z.B. Raum-, Zeit- und Personeninformationen mit geographischen, sachwort- oder personenbezogenen Informationssystemen verbinden.¹²⁵ Metadaten werden entweder direkt im gleichen DBMS verwaltet, in welchem auch die eigentlichen Textzeugen gespeichert werden, oder getrennt in spezialisierten Metadatenrepositorien besonders aus dem Bereich der Archive und Bibliotheken, wie der Flexible Extensible Digital Object and Repository Architecture (Fedora),¹²⁶ DSpace¹²⁷ oder EPrints¹²⁸. Diese zumeist innerhalb der Open Archive Initiative (OAI)¹²⁹ entstandenen Projekte sorgen für eine standardisierte Metadatenhaltung in unterschiedlichen Formaten wie dem Metadata Encoding and Transmission Standard (METS) oder dem Machine-Readable Cataloging (MARC), unterstützen damit einen genormten Zugriff auf die Daten und erleichtern den Austausch zwischen Forschungsprojekten, Bibliotheken und Archiven.

Die Retrievalfunktionalität der Digitalen Edition hängt im wesentlichen vom verwendeten DBMS ab und unterstützt zumeist die gängigen Grundfunktionen wie Volltext- oder kombinierte 'feldspezifische' Suchen inklusive Trunkierungsmechanismen. Je nach verwendeten Indexierungstools, DBMS-Erweiterungen oder zusätzlichen Analysewerkzeugen können weitere Funktionalitäten implementiert werden. Diese können z.B. sein: reguläre Ausdrücke, Phrasensuchen, Distanzsuchen (Wortabstände), Fuzzysearches¹³⁰, N-Gram-Indizes,¹³¹ Facetted Searches, Filterung der Suchergebnisse, Gewichtung von Suchtermen, Text-Mining-Verfahren etc. Browsingmechanismen vereinfachen den Zugang zu Quellen- und Kontextmaterialien und werden in unterschiedlichen, oft baumartigen Navigationsstrukturen realisiert. Die Art der Navigationsstruktur kann dabei recht unterschiedlich sein. Navigationsmechanismen über die verschiedensten semantischen Kategorien wie z.B. Schlagwörter, Personen, Daten, Archivfundstellen, geographische Kategorien, logische Strukturen wie Kapitel, Unterkapitel, Seiten etc. können als einfache Hierarchie, Baumstruktur oder Facettierte Navigation zur Verfügung stehen. Die Präsentation des Faksimile erfolgt in der Regel über einen der zahlreichen Image-Viewer, der die Grundfunktionen des Zoomens und des Pannings beherrscht. Proxybasierte Programmieretechniken erlauben zudem, hochauflösende Bilddaten dynamisch erst zum Zeitpunkt der Ausschnittvergrößerung nachzuladen, was zu einer wesentlichen Minimierung der Ladezeiten führt.¹³²

Erste Ansätze der Text-Image-Verlinkung haben schon recht früh Einzug in die Visualisierung Digitaler Editionen gehalten. Zuerst durch die zuvor bereits erwähnten

¹²⁵ Z.B. mit Hilfe von Browserwidgets wie timeline, timeplot <http://www.simile-widgets.org/>, timemap <http://code.google.com/p/timemap/> oder den Verknüpfungsmöglichkeiten von Personen- und Schlagwortdaten über die Gemeinsame Normdatei (GND). http://www.dnb.de/DE/Service/DigitaleDienste/LinkedData/linkeddata_node.html

¹²⁶ <http://fedoraproject.org/>

¹²⁷ www.dspace.de/

¹²⁸ www.eprints.org/

¹²⁹ www.openarchives.org/

¹³⁰ Z.B. phonetisch basierter Algorithmen wie Soundex oder Edit-Distance-Algorithmen wie Damerau-Levenshtein-Implementierungen

¹³¹ Diese sind besonders geeignet für Sprachen, die sich nicht durch Nichtwortzeichen in Wort-Tokens aufteilen lassen, wie z.B. das Chinesische.

¹³² So z.B. bei digilib <http://digilib.berlios.de/>

HTML-standardkonformen Methoden der verweissensitiven Graphiken, später auch durch fortschrittlichere Konzepte, wie etwa appletbasierte Lösungen. Eines der ersten Projekte, das eine solche Lösung präsentierte, war das Blakearchive¹³³. Sowohl das verwendete Image-Annotation-Tool Inote, als auch der Viewer (ImageSizer) waren auf der Basis von Java-Applets programmiert und erlaubten das Annotieren von Bildregionen und das Verknüpfen von diesen mit entsprechenden Textbausteinen.¹³⁴ Neuere Anwendungen werden eher mit Hilfe javascriptbasierter Techniken implementiert,¹³⁵ bei denen die in der Regel punktiert oder rechteckig Bildausschnitte markiert werden können, wie bei dem zuvor schon erwähnten Image-Viewer von digilib oder auch bei dem zeilenbasierten Annotationstool tile.¹³⁶ Andere Projekte implementieren solche Funktionalitäten auch des öfteren auf Basis von FLASHanwendungen wie z.B. im Alcalá Account Book-Projekt.¹³⁷

Parallelansichten von Transkriptionen, Faksimile, Lesetext, Übersetzung etc. wurden in frühen Digitalen Editionen for allem durch browsereigene Funktionen verwirklicht.¹³⁸ Parallel zu den Entwicklungen client- und serverseitiger Webprogrammiertechniken hat sich auch das Erscheinungsbild Digitaler Editionen im Verlauf der letzten 20 Jahre verändert, von eher statischen HTML-Seiten hin zu dynamischen Webanwendungen. Interaktivere Möglichkeiten kamen erst mit der Etablierung schon erwähnter Techniken wie Ajax und spezialisierten javascript-Frameworks, wie JQuery¹³⁹ oder YUI (Yahoo User Interface),¹⁴⁰ auf. Der Trend ging in die Richtung, dynamische und interaktive Funktionalität mehr und mehr auf die Clientseite zu verlagern. Zuvor lag der Schwerpunkt in der Regel auf der Serverseite. Mit Hilfe der Verbindung von HTML-Formularen und Techniken wie CGI, PHP, Java Server Pages, Servlets etc. wurde der Inhalt dynamisch serverseitig erstellt und HTML-Seiten als Response zurückgeschickt. Die Interaktion mit dem Benutzer auf diese Art und Weise hatte allerdings den großen Nachteil, dass jedes Mal die komplette Seite neu geladen werden musste oder die Inhalte schon beim ersten Übertragen mitgeliefert wurden und in versteckten HTML-Elementen zum späteren Gebrauch gespeichert wurden. Die erste Variante hatte den Nachteil, dass für kleinere Aktualisierungen der Ansicht (etwa das

¹³³ <http://www.blakearchive.org/blake/public/about/tech/index.html>

¹³⁴ Der Nachteil bei dieser Lösung bestand darin, das Image in fest vorgeschriebene Regionen einteilen zu müssen, eine Zentrumsregion und die sie umgebenden vier Regionen der Himmelsrichtungen.

¹³⁵ Es ist nicht genau ersichtlich, warum sich die Applet-Technologie nicht wirklich durchsetzen konnte. Wahrscheinlich hängt dies unter anderem damit zusammen, dass nicht alle Browser- und Betriebssystemhersteller bereit waren, das JDK von Sun nativ in ihre Software zu integrieren. Somit musste für die Ausführung eines Applets erst das Basispaket manuell installiert werden. Zudem gelten Applets im Allgemeinen als 'störanfälliger'.

¹³⁶ <http://mith.umd.edu/tile/>

¹³⁷ <http://archives.forasfeasa.ie/application.shtml>

¹³⁸ So heisst es in der Einleitung zum Blakearchiv etwa: „Note that these features on the Object View Page all display in separate windows [...]. This is an extremely important feature of the Archive, a premeditated aspect of its design. In order to use the Archive to its full potential, you should learn how to manipulate multiple windows on your computer's desktop, and also how to open new browser windows yourself for making various comparisons and cross-references“
<http://www.blakearchive.org/blake/public/about/tour/tour.html>

¹³⁹ <http://jquery.com/>

¹⁴⁰ <http://yuilibrary.com/>

Einblenden von textkritischen Informationen am Rande der Transkription) aufgrund der zustandslosen Natur von HTTP noch einmal die Anfrage an den Server geschickt werden musste und als Antwort die komplette HTML-Seite neu generiert wurde. Der Nutzer musste auf die Antwort warten, konnte in der Zeit keine weiteren Interaktionen ausführen, Einstellungen, wie etwa die der Scroll-Leisten wurden wieder auf den Ausgangszustand zurück gesetzt, der Benutzer musste also wieder manuell zu seiner letzten Textposition zurück navigieren und der Server wurde durch die erneute Abfrage zusätzlich belastet. Die zweite Möglichkeit war nur begrenzt praktikabel, da alle potentiell interessanten Informationen schon bei der Anfrage berücksichtigt werden mussten, was wiederum zu einer größeren Belastung des Servers führte und die zu übertragende Datenmenge erhöhte. Mit asynchronen Programmierstechniken, wie sie etwa Ajax¹⁴¹ bietet (Asynchronous JavaScript and XML), lassen sich einzelne Teile von Webseiten aktualisieren, ohne dabei die komplette Seite neu laden zu müssen. Damit werden die Nachteile des zustandslosen Hypertext Transfer Protocol umgangen. Inhalte vom Server können dynamisch nachgeladen und direkt in das Document Object Model eingefügt werden. So wird den Anforderungen des Benutzers Rechnung getragen, nur jeweils die Information neu zu laden, die er auch benötigt. Während der Client auf die Antwort wartet, kann die Anwendung auf weitere Interaktionen des Benutzers reagieren. Somit kann ein ähnliches Verhalten der Anwendung wie bei lokalen Desktopanwendungen simuliert werden.¹⁴² Dadurch wurde zugleich auch die Grundlage gelegt, einzelne Arbeitsschritte beim Erstellen von Digitalen Editionen ins Web zu verlagern. Zur Zeit wird der funktionale Unterschied zwischen browserbasierten Webanwendungen und Desktopanwendungen zusehends kleiner. Kontinuierlich steigende Übertragungsraten, asynchrone Technologien wie Ajax oder browserbasierte XML-Editoren¹⁴³ unterstützen diesen Trend. Welche Anwendungsbereiche das sind und woran es bei diesen in Bezug auf die Bearbeitung und Publikation Digitaler Editionen noch mangelt, soll im nächsten Kapitel beschrieben werden.

1.2.4 Software

1.2.4.1 Bildbearbeitung

Besonders historische Textüberlieferungen zeichnen sich häufig durch ihren schlechten Erhaltungszustand aus. Umweltbedingter Materialverfall des Beschreibstoffes wie Ausbleichen der Tinte, Pilzfraß, Wasser- oder Brandschäden etc. machen eine Vorverarbeitung des Bilddigitalisats meist unumgänglich¹⁴⁴ bzw. sorgen durch verschiedene Bildbearbeitungstechniken erst für die Lesbarkeit des Textes. Neben phototechnischen

¹⁴¹ <http://www.w3schools.com/ajax/default.asp>

¹⁴² Ein großer Unterschied besteht derzeit noch in der Übertragungszeit beim Datentransfer. Zum einen ist diese bei lokalen Anwendungen naturgemäß kürzer, zum anderen ist sie bei Webanwendungen stark von der Auslastung und Rechenleistung des übertragenden Servers abhängig. Letzteres hängt maßgeblich von der Auslegung des serverseitigen Systems ab.

¹⁴³ Wie z.B. Codemirror - <http://codemirror.net/>

¹⁴⁴ Julia CRAIG-MCFEELY: Digital Image Archive of Medieval Music: The evolution of a digital resource, in: Digital Medievalist 3 2008, URL: <http://www.digitalmedievalist.org/journal/3/mcfeely/>, §48ff.

Verfahren wie Multispektral- oder Fluoreszenzanalysen,¹⁴⁵ die bei der Rekonstruktion beschädigter Manuskripte vor allem in der Paläographie eine große Bedeutung haben, werden zur Vorverarbeitung in der Regel konventionelle Bildbearbeitungsprogramme wie Adobe Photoshop, Adobe Illustrator oder auch das unter der Gnu-Lizenz veröffentlichte Gimp verwendet.¹⁴⁶ Durch Farbwertkorrekturen oder dem Verwenden von Medianfiltern lässt sich bei Rastergraphiken das Bildrauschen unterdrücken. Schwellwertverfahren ermöglichen eine Trennung der Hintergrundinformation und der eigentlichen Schrift.¹⁴⁷ Bei der Rekonstruktion von kleineren Beschädigungen des Manuskripts im Text können z.B. Inpainting-Methoden angewendet werden.¹⁴⁸ Einen interessanten Ansatz verfolgt das von Alan Bowman initiierte eSAD-Projekt.¹⁴⁹ Das innerhalb dieses Projekts entstandene ISS-Interpretation Support System¹⁵⁰ versucht dabei, die Arbeitsweise von Paläographen bei der Rekonstruktion beschädigter Dokumente zu analysieren, die daraus gewonnenen Prinzipien zu verallgemeinern und auf der Softwareebene zu implementieren.¹⁵¹ Eine genauere Beschreibung des Systems ist im folgenden Abschnitt zu finden. Ein weiteres wegweisendes System in diesem Zusammenhang war das aus den Arbeiten zur Digitalen Edition 'Electronic Beowulf'¹⁵² entstandene EPPT (Electronic Production & Presentation Technology), das aber anscheinend seit 2009 nicht mehr weiterentwickelt wurde.

1.2.4.2 Texterkennung

Im Bereich der Digitalen Editionen lassen sich im wesentlichen zwei Bereiche der Texterkennung unterscheiden. 1. die Retrodigitalisierung bereits gedruckt vorliegender Editionen und Textzeugen und 2. die handschriftliche Texterkennung. Im ersten Fall kommen vor allem OCR/ICR-basierte Systeme wie Abbey-FineReader¹⁵³ oder OmniPage¹⁵⁴ zum Einsatz. Die gescannte Rastergraphik wird bei diesen Systemen in

¹⁴⁵ Melanie GAU/Robert SABLATNIG: Technical Approaches To Manuscript Analysis and Reconstruction, in: Universität Wien 2007, S. 1–41, URL: http://www.zde.uni-wuerzburg.de/fileadmin/04100700/_temp_/GauSablatnig.pdf, 19ff.

¹⁴⁶ Z.B. bei Keating: „Despite illuminating the manuscript with fluorescent cold lighting the brightness of the images varied because of changing ambient light within the library. The images therefore required post-processing to provide uniform brightness across the pages and collection. This post-processing, together with centering and cropping, was accomplished using standard image manipulation software.“ G. KEATING u. a.: Account Book (wie Anm. 87), §14.

¹⁴⁷ Vgl. Melanie GAU u. a.: Image Acquisition and Processing Routines for Damaged Manuscripts, in: Digital Medievalist 6 2010, URL: <http://www.digitalmedievalist.org/journal/6/gau/>, §18ff oder auch Jan Gerrit WIENERS: Zur Erweiterungsfähigkeit bestehender OCR Verfahren auf den Bereich extrem früher Drucke (Magisterarbeit), Universität zu Köln 2008, URL: <http://old.hki.uni-koeln.de/studium/MA/>, S. 26.

¹⁴⁸ GAU/SABLATNIG: Approaches (wie Anm. 145), S. 32.

¹⁴⁹ <http://esad.classics.ox.ac.uk/>

¹⁵⁰ Henriette ROUED-CUNLIFFE: Towards an Interpretation Support System for Reading Ancient Documents, in: Literary and Linguistic Computing. Centre for the Study of Ancient Documents 2010, S. 1–15, URL: http://www.academia.edu/1786977/Towards_an_Interpretation_Support_System_for_Reading_Ancient_Documents.

¹⁵¹ <http://www.ahessc.ac.uk/image-text-interpretation>

¹⁵² <http://ebeowulf.uky.edu/studyingbeowulfs/criticaledition>

¹⁵³ www.abbyy.com/

¹⁵⁴ <http://www.nuance.com/for-business/by-product/omnipage/index.htm>

der Regel einem Vorverarbeitungsprozess unterzogen, in dem Stör- und Rauschunterdrückungsverfahren, Graustufenumwandlungen zur Reduktion der Farbinformation, Rangordnungsfilter etc. angewendet werden, um die Fehlerquote der daran anschließenden Verfahren der Textsegmentierung und Zeichenerkennung nach Möglichkeit zu minimieren. Mittels unterschiedlicher Analyseverfahren wie der Farb- und Layouterkennung und Korrekturverfahren wie der Skew-Korrektur werden danach Textblöcke selektiert und alle anderen Bereiche der Graphik, die nicht als solche erkannt wurden, von den weiteren Analyseprozessen ausgeschlossen. In der Regel schließt sich an diesen Prozess die Umwandlung in ein bitonales Bild mittels unterschiedlicher Schwellwert-Algorithmen an, um die eigentliche Texterkennung zu ermöglichen. Bei dieser werden Algorithmen angewendet, die die Textblöcke in Zeilen, Wörter und Zeichen segmentieren. Die so ermittelten Wörter werden dann z.B. morphologisch analysiert und mit vorhandenen Wörterbucheinträgen verglichen. Die Erkennungsgenauigkeit ist bei qualitativ guten Druckvorlagen recht hoch, weswegen sich die genannten Softwarepakete großer Beliebtheit bei der Retrodigitalisierung erfreuen.¹⁵⁵ Für extrem frühe Druckerzeugnisse stellt Wieners ein Verfahren vor, mit dem Glyphen identifiziert und somit anhand neuronaler Netze Cluster erstellt werden können, die in Form einer Wissensbasis die Grundlage für weitere automatische Schriftzeichenrcknungen bilden können.¹⁵⁶

Bei der Handschriftlichen Texterkennung werden im Prinzip die gleichen Verfahrensschritte durchlaufen, wobei sich aber vor allem der Prozess, der sich der Vorverarbeitung anschließt, also die Objekterkennung und deren Segmentierung in Zeilen, Wörter und Zeichen, ungleich schwieriger gestaltet. Besonders im Bereich der Digitalisierung von Handschriften führte dies zu reger Forschungstätigkeit und der Erprobung unterschiedlicher Methoden.¹⁵⁷ Melissa Terras beschreibt ein solches System am Beispiel altrömischer Majuskelskursive, das auf dem Agent-System Grava¹⁵⁸ basiert und Paläographen beim Entziffern fragmentarisch erhaltener Handschriften

¹⁵⁵ Abbey-FineReader ist zur Zeit wohl die am häufigsten verwendete OCR-Software im Bereich der Retrodigitalisierung (abgesehen von Initiativen, die mit Hilfe von Google vor allem größere Archiv- und Bibliotheksbestände digitalisieren, wie z.B. die Bayrische Staatsbibliothek). Auf Projektbasis stellt Abbey-FineReader ein OCR-Modul zur Verfügung, welches auch Dokumente aus dem Zeitraum zwischen dem 17ten und 20ten Jahrhundert verarbeiten kann. Die Erkennungsgenauigkeit liegt nach eigenen Angaben bei ca. 98%. Da keine eigenen Tests durchgeführt wurden, kann hier nur auf Erfahrungsberichte verwiesen werden, die dies, mit einigen Einschränkungen, in der Regel bestätigen. Siehe hierzu: [<http://www.iais.fraunhofer.de/nzz.html>], [<http://www.ub.uni-dortmund.de/listen/inetbib/msg32677.html>] oder [<http://neuemedien.uni-graz.at/modules941e.html?name=Content&pa=showpage&pid=13>]

¹⁵⁶ WIENERS: OCR (wie Anm. 147).

¹⁵⁷ Einführungen hierzu bieten z.B. R.G. CASEY/E. LECOLINET: *Strategies in character segmentation: A survey*, in: 3rd International Conference on Document Analysis and Recognition 1995, S. 1028–1033 oder W. SENIOR: *Off-line handwritten recognition: A review and experiments*, in: Technical Report CUED/F-INFENG/TR105 1992, S. 1028–1033.

¹⁵⁸ Paul ROBERTSON/Robert LADDAGA: *The GRAVA Self-Adaptive Architecture: History, Design, Applications and Challenges*, in: Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference 2004, S. 298–303, URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.2.1328&rep=rep1&type=pdf>.

unterstützt.¹⁵⁹ Das System arbeitet mit Methoden der Feature-Extraction¹⁶⁰ und unterteilt drei Phasen der Texterkennung. Auf der Ebene der ‘low level features’ identifiziert es einzelne Bestandteile von Zeichen (strokes) und gewinnt daraus Informationen über die Länge, Position und Ausrichtung eines Zeichens. Die zweite Ebene der character-Level vergleicht auf der Grundlage eines so gewonnenen Trainings-Korpus von Alphabetzeichen unbekannte Glyphen mit den aus Level 1 gewonnenen Informationen. Zum Mustervergleich werden auch hier wieder Methoden aus dem Bereich der AI-Forschung angewandt. Auf dem word-Level werden die durch Level 1 und 2 gefundenen Schriftzeichen zu Zeichenketten kombiniert und mit Wörterbucheinträgen verglichen. Solche Verfahren werden nicht nur zur Texterkennung verwendet, sondern beispielsweise auch in der Paläographie zu Handschriftenanalysen herangezogen. Hierbei werden Feature-Extraction-Methoden implementiert, die charakteristische Merkmale einer Handschrift extrahieren und somit als Vergleichsbasis für fremde Handschriften dienen können.¹⁶¹

Das Hauptproblem solcher Methoden zur Texterkennung auf Buchstabenebene liegt in der Segmentierung in einzelne Zeichen. Fließende Handschriften, die nur sehr schwer eine Unterteilung anhand von eindeutigen Buchstabengrenzen erlauben, lassen sich bis heute nicht zufriedenstellend erkennen. Methoden, die zumindest die Segmentierung von solchen Handschriften in Zeilen und auf Wortebene ermöglichen, beschreibt z.B. Feineis.¹⁶² Da all diese Systeme in der Regel mit Trainingssets arbeiten, gilt es bei Editionsprojekten abzuwägen, ob der Aufwand zur Erstellung solcher Sets die Verwendung dieser Systeme rechtfertigt. Wie in Kapitel 3.3.3.2 noch gezeigt wird, bieten sich automatische Methoden der Zeilen- und Wortsegmentierung auch im Umfeld browserbasierter Anwendungen an, um genaue diplomatische Abschriften und Textüberblendungen zu realisieren. Lässt sich keine dieser Methoden sinnvoll einsetzen, bleibt nur die manuelle Erfassung, wie das simple Abschreiben oder das Double-Keying-Verfahren, bei dem der Text von zwei Personen abgeschrieben wird und die Abweichungen meist maschinell erfasst und wiederum manuell korrigiert werden. Dieses Verfahren zeichnet sich besonders durch seine geringe Fehleranfälligkeit aus.¹⁶³

¹⁵⁹ Hierbei handelt es sich um das im vorigen Kapitel schon erwähnte ISS-Interpretation Support System. Vgl. Melissa M. TERRAS: *Image to Interpretation. An Intelligent System to Aid Historians in Reading the Vindolanda Texts*, New York 2006.

¹⁶⁰ Als Einführung siehe Joutel GUILLAUME u. a.: *Curvelets based Feature Extraction of handwritten shapes for ancient manuscripts classification*, in: *Laboratoire d'Informatique en Image et Systèmes d'information* 2007, S. 1–12, URL: liris.cnrs.fr/Documents/Liris-2688.pdf.

¹⁶¹ Vgl. Florence CLOPPET: *New Tools for Exploring, Analysing and Categorising Medieval Scripts*, in: *Digital Medievalist* 7 2011, URL: <http://www.digitalmedievalist.org/journal/7/cloppet/> oder auch Peter A. STOKES: *Palaeography and Image-Processing: Some Solutions and Problems*, in: *Digital Medievalist* 3 2008, URL: <http://www.digitalmedievalist.org/journal/3/stokes/>.

¹⁶² Markus FEINEIS: *Wortgenaue Annotation digitalisierter mittelalterlicher Handschriften*, Bayerische Julius-Maximilians-Universität Würzburg. Institut für Informatik 2008.

¹⁶³ Vgl. hierzu auch Susanne HAAF/Frank WIEGAND/Alexander GEYKEN: *Measuring the Correctness of Double-Keying: Error Classification and Quality Control in a Large Corpus of TEI-Annotated Historical Text*, in: *Journal of the Text Encoding Initiative* 2013, URL: <http://jtei.revues.org/739>.

1.2.4.3 Kollationierung

Das Kollationieren geht in der Editionswissenschaft dem Prozess der Emendation voraus und bildet die Grundlage textkritischer Überlegungen in Bezug auf Abstammungs- und Verwandtschaftsverhältnisse unterschiedlicher Textzeugen. Bei identischen Abweichungen von mehreren Textzeugen (im Verhältnis zu einer früheren Textüberlieferung) kann so z.B. darauf geschlossen werden, dass diese Texte in einem direkten Bezug zueinander stehen (z.B. ein Text von dem anderen abgeschrieben wurde), da es unwahrscheinlich ist, dass diese Abweichungen unabhängig voneinander entstanden sind. Auf Grundlage dieses Textvergleichs wurde in der traditionellen Editionswissenschaft oft die Entscheidung getroffen, welcher Text zum Basistext der Edition werden sollte und welche Texte als Varianten in den kritischen Apparat verbannt wurden. Das manuelle Kollationieren ist ein extrem aufwändiger Prozess und so hat es schon relativ früh Bestrebungen gegeben, diesen durch technologische Lösungen effizienter zu gestalten. Als einer der Pioniere auf diesem Gebiet gilt bis heute Charlton Hinman, der in den späten 40er Jahren den nach ihm benannten Hinman-Collator entwickelte.¹⁶⁴ Mit Hilfe dieses auf einem optomechanischen Verfahren basierenden Gerätes war es möglich, variante Textstellen in unterschiedlichen Druckvorlagen zu identifizieren. Da dieses Verfahren mit Überblendungstechniken arbeitet, war es allerdings nur zum Identifizieren kleinerer Textabweichungen geeignet.¹⁶⁵

Aufgrund des algorithmisch gut abbildbaren Vergleichs von Zeichenketten sind Verfahren zum automatischen Kollationieren auch schon sehr früh in der elektronischen Editionswissenschaft angewendet worden. Tustep etwa verfügte schon seit den 70er Jahren über solche Möglichkeiten. Heutzutage existieren eine Reihe solcher Werkzeuge, um sowohl den Prozess des eigentlichen Kollationierens unterschiedlicher Texte zu automatisieren, als auch den anschließenden Vergleich varianter Textstellen zu erleichtern.¹⁶⁶ Das aus Collate hervorgegangene CollateX etwa oder das an der University of Virginia entwickelte Juxta, das auch als Web-Applikation zur Verfügung steht. CollateX ist beispielsweise als Zusatzmodul im TextGridLab integriert und unterstützt zusätzlich mit dem GraphML-Format¹⁶⁷ einen Export zur graphenbasierten Darstellung und Weiterverarbeitung. Juxta ist seit 2012 auch in einer ersten Betaversion als Webservice implementiert. Beide Formate unterstützen den Export der kollationierten Daten in einen TEI-Output mit Apparat-Elementen im parallel-segmentation-mode. Somit lassen sich anschließend Tools wie die Versioning Machine¹⁶⁸ nutzen, einem Modul zum parallelen Betrachten und Vergleichen von varianten Textzeugen. Ein großes

¹⁶⁴ Charlton HINMAN: Mechanized Collation at the Houghton Library, in: Library Bulletin 9 1955, S. 132–134.

¹⁶⁵ <http://www.uni-muenster.de/Buchwiss/Organisation/Rosenberg.html>

¹⁶⁶ Kollationsalgorithmen bilden die Grundlage für die automatische Verifizierung von Varianten. Einen guten Überblick über die Problematiken der unterschiedlichen Klassen von Algorithmen bietet Schmidt. Desmond SCHMIDT: Merging Multi-Version Texts: a General Solution to the Overlap Problem, in: Balisage Series on Markup Technologies 3 2009, URL: <http://www.balisage.net/Proceedings/vol3/html/Schmidt01/BalisageVol3-Schmidt01.html>. Vor allem das Problem der automatischen Erkennung von Transpositionen wird hier ausführlich erläutert.

¹⁶⁷ <http://graphml.graphdrawing.org/>

¹⁶⁸ <http://v-machine.org/file.php>

Problem beim automatischen Kollationieren besteht darin, dass die Aussage, was eine textliche Variante ist, nicht immer ohne Kontextwissen getroffen werden kann.

Beispiel 1:

Text a: Das große weiße schöne Haus

Text b: Das weiße weiße Haus

führt bei Juxta zu dem Ergebnis in Listing 1.1. Hier wird folgende Sicht präferiert (weiße als Variante zu schöne):

Text a: Das große weiße schöne Haus

Text b: Das weiße weiße Haus

Es ist aber auch durchaus denkbar, dass weiße eine Variante von große ist:

Text a: Das große weiße schöne Haus

Text b: Das weiße weiße Haus

```

1 <p>
2   Das
3   <app>
4     <rdg wit="#wit-a">große</rdg>
5     <rdg wit="#wit-b"></rdg>
6   </app> weiße
7   <app>
8     <rdg wit="#wit-a">schöne</rdg>
9     <rdg wit="#wit-b">weiße</rdg>
10  </app> Haus<lb/>
11 </p>

```

Listing 1.1

Sollte eine Entscheidung nicht anhand von anderen kontextabhängigen Kriterien, wie z.B. einer Aussage des Autors getroffen werden können, würde man wohl am ehesten die komplette Adjektivreihe oder die ganze Zeile als Lesart interpretieren:

```

1 <p>
2   Das
3   <app>
4     <rdg wit="#wit-a">große weiße schöne</rdg>
5     <rdg wit="#wit-b">weiße weiße</rdg>
6   </app>
7   Haus
8 </p>

```

Listing 1.2

Werden die Varianten komplexer und befinden sie sich zudem nicht an der gleichen Stelle im Text, kann dies zu unschönen Effekten führen, wie in folgendem Beispiel zu sehen.

Beispiel 2:

Text a) Die Welt steht still
so wie sie will

Die weißen Schimmel
reiten durch den Schnee

Text b) Die schwarzen Rappen
reiten durch den Schnee

Die Welt steht still
so wie sie will

Der Output bei Juxtapos hierzu sieht wie folgt aus:

```

1  <p>
2  Die
3  <app>
4      <rdg wit="#wit-a"> </rdg>
5      <rdg wit="#wit-b">schwarzen Rappen<lb/>reiten durch den Schnee<lb/><lb/
      >Die</rdg>
6  </app>
7  Welt steht still <lb/>
8  so wie sie will<lb/>
9  <lb/>
10 <app>
11     <rdg wit="#wit-a">Die weißen Schimmel<lb/>reiten durch den Schnee</rdg>
12     <rdg wit="#wit-b"></rdg>
13 </app>
14 </p>

```

Listing 1.3

Als Varianten der durch den Schnee reitenden Pferde werden hier jeweils für die beiden anderen Textzeugen leere rdg-Elemente angegeben. Es wird also nicht erkannt, dass die beiden Zeilen in einem Variantenverhältnis zueinander stehen.¹⁶⁹ Durch die Vertauschung der beiden Zweizeiler und der zusätzlichen Variante ‘schwarzen Rappen’ anstelle von ‘weißen Schimmel’ ist der Algorithmus nicht mehr in der Lage, zu

¹⁶⁹ Hinzu kommt, dass das ‘Die’ von ‘Die Welt steht still’ in Zeile 4 mit in den Apparat geschrieben wurde.

erkennen, dass es sich um zwei Varianten ein und derselben Strophe handelt, deren Abfolge sich durch eine Transposition verändert hat. Hier würde man eher eine Struktur wie in Listing 1.4 erwarten (wenn Text a als Ausgangstext angenommen wird). Es müssten also Ankerpunkte gesetzt werden, die signalisieren, welche Textabschnitte miteinander verglichen werden sollen oder semantische Analysen während des Kollationierungsprozesses berücksichtigt werden. Bei den genannten Systemen fehlt es an umfangreichen Konfigurationsmöglichkeiten. Grundsätzlich muss in solchen Situationen projektabhängig geprüft werden, inwieweit sich der Einsatz automatischer Kollationierung lohnt. Immerhin müssen alle Textzeugen für einen solchen automatischen Vergleich schon digital erfasst worden sein. Bei nur gering abweichenden Textzeugen kann dieser Aufwand den Nutzen übertreffen, insbesondere in Fällen, in denen die Art der Überlieferung keine automatische Texterkennung zulässt und die Erfassung manuell von von statten gehen muss.¹⁷⁰

```

1 <p>
2   <span xml:id="ib1">
3     Die
4     <app>
5       <rdg wit="#wit-a">weißen Schimmel</rdg>
6       <rdg wit="#wit-b">schwarzen Rappen</rdg>
7     </app><lb/>
8     reiten durch den Schnee
9   </span>
10  <span xml:id="ib2">
11    Welt steht still <lb/>
12    so wie sie will<lb/>
13  </span>
14 </p>
15 ...
16 <transpose wit="#wit-b">
17   <ptr target="#ib02"/>
18   <ptr target="#ib01">
19 </transpose>

```

Listing 1.4

¹⁷⁰ Schubert zufolge könne es z.B. auch „effektiver sein, Übertragung, Vergleich und Bewertung gleichzeitig (manuell) auszuführen. Hier können sich vorab stemmatologische Beziehungen offenbaren, die die vollständige oder teilweise Transkription einer Vorlage unnötig erscheinen lassen, weil sie mit einer anderen hinreichend parallel läuft.“ <http://hsozkult.geschichte.hu-berlin.de/tagungsberichte/id=1157>

1.2.4.4 Datenerfassung/-bearbeitung

Aufgrund der spezifischen Anforderungen an Softwaresysteme haben sich in der Editions-wissenschaft eine Reihe von Werkzeugen etabliert, die entweder im besonde-ren einzelne Editions-methoden und Fachrichtungen unterstützen oder versuchen, mit Komplettsystemen umfassende Lösungsansätze für den gesamten Editionsprozess zur Verfügung zu stellen. Zu Beginn waren es im Bereich der semantischen markuporien-tierten Texterschließung aber vor allem herkömmliche XML-Editoren. Ihr Vorteil war und ist, dass sie spezialisiert sind auf die XML-Verarbeitung, ihr Nachteil, dass sie nicht auf die spezifischen Bedürfnisse der Editions-wissenschaft eingehen. In der Regel erlauben sie schemakonforme Validierung, unterschiedliche Bearbeitungs- und Naviga-tionsmodi wie z.B. baumartige Darstellungen der Dokumentstruktur oder auch direk-tes Edieren von serverseitigen XML-DBMS-Inhalten mittels geeigneter Schnittstellen, wie z.B. bei der Kombination von oXygen¹⁷¹ und eXist-db. Einer ihrer wesentlichen Nachteile besteht darin, dass die Texte bei zunehmender Erschließungstiefe deutlich an Lesbarkeit¹⁷² verlieren und in der Regel keine Möglichkeiten bestehen, zusätzliche Materialien oder Arbeitsschritte direkt in den Erfassungsprozess zu integrieren, wie z.B. die Verknüpfung textueller Informationen mit Bilddigitalisaten und deren Inhal-ten, die Verwendung von Kollationierungsmodulen oder die Integration externer In-formationenquellen wie Fachwörterbücher, Personen- oder Schlagwortverzeichnisse etc. Aufgrund der eingeschränkten Lesbarkeit von XML-Daten geht hier der Trend seit geraumer Zeit wieder in die Richtung klassischer wysiwyg-Prinzipien (oder zumindest parallel auch solche Erfassungs- und Darstellungsoptionen zu ermöglichen). Oxygen beispielsweise bietet ein authoring-tool an, mit dem die Eingabe der Daten mit Hilfe von css-Stylesheets formatiert und auf ähnliche Art und Weise bearbeitet werden kann wie bei browserbasierten RTE-Anwendungen (Rich Text Editoren), die häufig in CMS-Systeme integriert sind. Im Bereich der Spezialsysteme sind insbesondere Desktopan-wendungen, die aus Einzelprojekten entstanden sind. Zu nennen sind hier neben dem bereits genannten Tustep vor allem ältere Projekte, wie z.B. der seit 1997 an der Ös-terreichischen Akademie der Wissenschaften in Wien entwickelte Classical Text Edi-tor (CTE), der die Datenerfassung nach dem wysiwyg-Prinzip unterstützt, allerdings ähnlich wie Tustep als primäres Ziel noch immer auf die klassische Printeditionen setzt, EPPT oder auch neuere Systeme, wie das am Trierer Kompetenzzentrum für

¹⁷¹ <http://www.oxygenxml.com/>

¹⁷² Stellvertretend für diese Kritik an klassischen XML-Editoren sei auf die Überlegungen Ritters zur Entwicklung alternativer Erfassungsmöglichkeiten für die Edition der Dramen von Karl Ferdinand Gutzkow hingewiesen: „Üblicherweise schließt sich an die Dokumentenanalyse eine Erfassung des Korpus mit Hilfe eines gängigen XML-Editors an. Diese Programme ermöglichen schemakonforme Eingaben und garantieren durch die implizite Konsistenzprüfung valide und wohlgeformte Dokumente. Je komplexer die Textstruktur, desto mehr Markups enthalten die Dokumente, so dass der edierte Text in den Hintergrund rückt. Wünschenswert wäre jedoch eine Arbeitsumgebung, die den eigentlichen Inhalt in den Vordergrund stellt. [...] Die Auszeichnung von Textpassagen sollte wie bei der Formatierung in konventionellen Textverarbeitungssystemen erfolgen, wobei die XML-Auszeichnungen für die Bearbeiter durch modifizierten Schriftstil beziehungsweise durch veränderte Schriftart oder -farbe angezeigt werden, ohne dass das Mark-up selbst sichtbar wird.“ RITTER/SCHÜTZ/TEITGE: Dramen von Karl Ferdinand Gutzkow (wie Anm. 83).

elektronische Erschließungs- und Publikationsverfahren in den Geisteswissenschaften entstandene Editionstool Forschungsnetzwerk und Datenbanksystem (FUD¹⁷³), das ganz nach dem klassischen relationalen Prinzip verwaltet und unterschiedliche Eingabemasken für Transkriptionen, Metadaten, Übersetzungen etc. bereit stellt, innerhalb dieser Masken aber auch XML-Annotationen unterstützt. Die Kritik an solchen Systemen besteht vor allem in der mangelnden Interoperabilität, einer zu großen Spezialisierung, unnötigen Mehrfachentwicklungen und mangelnder Nachnutzbarkeit.¹⁷⁴

Ein gutes Beispiel für ein System, das auf diese Kritik mit dem Anspruch reagierte, auf allen Ebenen fächerübergreifend die Arbeit an einer Digitalen Edition zu unterstützen und wiederverwendbare Werkzeuge zur Verfügung zu stellen, ist das Textgrid-Projekt.¹⁷⁵ Aus diesem Projekt ist unter dem Namen TextgridLab eine Sammlung von Werkzeugen entstanden, die entweder aus Eigenentwicklungen bestehen, wie z.B. einem auf Eclipse basierenden Editor mit Text-Text- und Text-Image-Linking-Modulen oder die Integration externer Anwendungen wie oXygen, Digilib oder das an der Berlin Brandenburger Akademie der Wissenschaften entwickelte Publikationsframework Sade ermöglichen. Darüber hinaus sind auch Werkzeuge und Analysetools aus den Fachbereichen der Philologie (CollateX), Linguistik (Lexus, Cosmas II, Annex) oder der Musikwissenschaft (MIE Score Editor) Bestandteil der Entwicklungsumgebung. Solche Entwicklungen tragen dem Trend Rechnung, die Arbeitsumgebungen von Editionsprojekten mehr und mehr ins 'Netz' zu verlagern und werden auch unter dem Begriff 'Virtual Research Environments' (VRE) subsumiert.¹⁷⁶

Ein weiterer aktueller Trend geht in die Richtung, das kollaborative Edieren von Datenbeständen direkt in browserbasierten Umgebungen zu ermöglichen.¹⁷⁷ Die Grund-

¹⁷³ <http://fud.uni-trier.de/>

¹⁷⁴ Hierzu stellvertretend Aschenbrenner: „Forscherguppen investieren oft viel Energie in die Entwicklung von Werkzeugen, die auch in anderen Kontexten nützlich wären. Aber doch werden in der Praxis viele Werkzeuge mehrfach entwickelt, weil oftmals die technischen Schnittstellen fehlen, keine Interoperabilität der Datenformate gegeben ist, und die Werkzeuge auf eine spezialisierte Forschungsfrage ausgerichtet sind. Die Interoperabilität von Werkzeugen durch Standards und entsprechende technische Umgebungen zu fördern, ist ein Anliegen vieler der betrachteten Infrastrukturaktivitäten.“ Andreas ASCHENBRENNER/Katja MEFFERT: Wissenschaftliche Infrastruktur in den Geisteswissenschaften? – Eine Wegbeschreibung, in: Jahrbuch für Computerphilologie 9 2009, URL: <http://computerphilologie.digital-humanities.de/jg07/aschmeff.html>, §52 oder auch Schreibmann: „And despite years of discussions between the cultural heritage and scholarly communities, there are still too few examples of successful partnerships that allow each community to bring their strengths to collaborative digital projects“ Susan SCHREIBMAN: The Text Encoding Initiative. An Interchange Format Once Again, in: Jahrbuch für Computerphilologie 10 2008, URL: <http://computerphilologie.digital-humanities.de/jg08/schreibman.html>, §11.

¹⁷⁵ <http://www.textgrid.de/>

¹⁷⁶ Zu den Vorteilen solcher Systeme stellvertretend Bowman: „to 'work in different countries on the same text' simultaneously, allowing researchers to collaborate and form 'virtual gatherings' without being unnecessarily constrained by time or travel.“ Alan K. BOWMAN u.a.: A Virtual Research Environment for the Study of Documents and Manuscripts, in: Digital Research in the Study of Classical Antiquity 2010, S. 87–103, hier S. 88 VRE: „The purpose of a Virtual Research Environment (VRE) is to help researchers from all disciplines to work collaboratively by managing the increasingly complex range of tasks involved in carrying out research on both small and large scales.“ [<http://misc.jisc.ac.uk/vre/>]

¹⁷⁷ Diese Entwicklung ist stark durch Projekte aus dem Bereich der zeitweise unter dem Begriff

idee solcher Systeme besteht darin, auf die freiwillige Mitarbeit von größeren Nutzergruppen (Crowdsourcing) zu setzen, um diese Ressourcen der Netzgemeinde effizienter nutzbar zu machen. Oft sind es fachspezifische Editionsplattformen die solche Methoden zu etablieren versuchen, wie z.B. das *monasterium*-Projekt, ein virtuelles Archiv mittelalterlicher und frühneuzeitlicher Urkunden, das Freiwillige dazu einlädt, an der Erschließung der Dokumentbestände mitzuwirken. Ein anderes Beispiel ist die Papyrusurkunden-Plattform *papyri.info*¹⁷⁸, die mit dem Recherchemodul *Papyrological Navigator* und dem *Papyrological Editor* Werkzeuge zur Verfügung stellt, um auf unterschiedliche Papyrus-Datenbestände zugreifen und diese transkribieren zu können. Aus dieser Art von Projekten entstehen immer wieder innovative onlinebasierte Werkzeuge zum Annotieren der Daten, wie der aus dem *monasterium*-Projekt hervorgegangene *Edit-Mom-Editor* oder der *Papyrological Editor*. Bei derartigen Systemen spielt die Usability eine zentrale Rolle, da die freiwilligen Editoren nicht durch aufwendige Schulungen in Spezialsysteme eingeführt werden können. So stellt *Edit-Mom* ein Annotationssystem zur Verfügung, um eine einfache und intuitive Datenerfassung zu ermöglichen und gleichzeitig alle XML-Notationen zu verbergen, die den Lese- und Bearbeitungsfluss des Benutzers behindern würden. Der *Papyrological Editor* unterstützt zusätzlich eine Versionshistorie und lässt die Bearbeitungen auch mit einer speziellen Kurznotation (*Leiden+*) zu, ein weiterer Versuch, den Aufwand, den die XML-Annotierungen oft verursachen, in Grenzen zu halten. Formattechnisch stützen sich diese Projekte meist auf die Richtlinien der TEI bzw. verwandter Auszeichnungsschemata wie im Falle von *Edit-Mom* auf die CEI oder im Papyrusurkunden-Projekt auf *EpiDoc*. Die Qualitätskontrolle wird dabei entweder durch die Benutzer selbst oder wie im Falle des *monasterium*-Projekts durch ein stufenbasiertes Redaktionssystem sichergestellt. Die Meinungen gehen hierbei durchaus auseinander, ob eine derartige Kontrolle unabdingbar für die Qualitätssicherung ist oder ob man die Schar der freiwilligen Helfer nicht unterschätzt, wie z.B. Gibbs bemerkt: „To assume that all work must be vetted by a firm editorial voice is to ignore the vast potential of highly trained and motivated community practitioners who want to work together to discover relevant texts.“¹⁷⁹

Weitere Möglichkeiten für das kollaborative Arbeiten in Projekten mit begrenztem Forscherkreis stellen inzwischen auch kommerzielle Softwaresysteme zur Verfügung. Diese Arbeitsumgebungen werden auch als *Groupware* bezeichnet und können für den studentischen oder universitären Gebrauch inzwischen kostenfrei lizenziert werden.¹⁸⁰ Meist unterstützen entsprechende Systeme eine große Anzahl von Dateiformaten und Multimediaobjekten. Kalenderfunktionen, automatische Benachrichtigungs-

Web 2.0 firmierenden interaktiven und kollaborativen Systeme wie z.B. der online-Enzyklopädie Wikipedia beeinflusst worden.

¹⁷⁸ <http://papyri.info>

¹⁷⁹ GIBBS: *Community Transcription* (wie Anm. 22), S. 14. Zusätzlich schlägt er ein System vor, das eine verteilte Versionskontrolle unterstützt: „it does not require that everyone must work with the same version of the document at the same time. People can work on different parts independently, sharing or not sharing work as they go“ ebd., S. 23.

¹⁸⁰ Ein Beispiel für eine solche kostenfreie Studentenversion ist unter <http://www.teamspace.de/> zu finden.

systeme, abgestufte Benutzergruppenverwaltung, Versionierungssysteme und Volltextsuchfunktionen unterstützen die projektbezogene Arbeitsorganisation. Der Nachteil dieser Systeme liegt allerdings darin, dass sie in der Regel keine spezifischen Funktionen zum Edieren geisteswissenschaftlichen Quellenmaterials bieten.

Aktuell dominieren Bestrebungen, virtuelle Forschungsumgebungen zu etablieren, die innerhalb größerer Verbund-Initiativen entstehen und mit erheblichen nationalen und europäischen Fördermitteln finanziert werden, wie z.B. Dariah,¹⁸¹ Clarin¹⁸² oder das innerhalb des Enrich-Projekts¹⁸³ entstandene Manuscriptorium (Europäische Digitale Bibliothek von Handschriften mit mehr als fünf Millionen digitalen Abbildungen¹⁸⁴). Ziel dieser Projekte ist es, eine nachhaltige Infrastruktur für die geisteswissenschaftliche Forschung aufzubauen, um die Kommunikation unter den Wissenschaftlern zu intensivieren, verteilte Datenbestände durch informationstechnologische Methoden verfügbar zu machen und einen Austausch von Methoden und Technologien verschiedener Fachrichtungen zu stärken.

Neben den textbasierten Annotationssystemen spielen die Möglichkeiten der Verlinkung von Textdaten und Bilddaten eine zunehmende Rolle bei der Transkription von Textzeugen.¹⁸⁵ Beeinflusst durch editionstheoretische Diskussionen, das Original wieder ins Zentrum des Interesses zu stellen,¹⁸⁶ sind es gerade die Bilddigitalisate, die das objektivste Abbild des Originals bieten können. Nur an ihnen lassen sich die editorischen Eingriffe überprüfen.¹⁸⁷ Eine einfache Beigabe des Digitalisats in Form einer hochauflösenden Bitmap-Graphik reicht hier allerdings nicht immer aus. Die Transkription ist Deutung des Befunds und in diesem Sinne auch stets nur eine mögliche Interpretation des materiellen Originals. Um die einzelnen Interpretationen genauestens auf das Original beziehen zu können, sollten in letzter Konsequenz alle interpretierten Textphänomene direkt im Original sichtbar gemacht werden. Dazu

¹⁸¹ <http://www.dariah.eu/>

¹⁸² <http://www.clarin.nl/>

¹⁸³ <http://enrich.manuscriptorium.com/index.php?q=how-to-join-us>

¹⁸⁴ <http://www.manuscriptorium.com>

¹⁸⁵ Darauf, dass es hierfür noch an einer eigenen Methodik fehlt, verweist schon Sahle: „Von Editionen, die heute keine digitalen Abbildungen begeben, kann man eine explizite Begründung für diesen Verzicht erwarten. In welcher Form digitale Abbildungen beigegeben oder mit der Edition verknüpft werden, wie sie in die Oberflächengestaltung und Navigationsstrukturen der Edition eingebunden und wie sie mit den anderen Formen der Textwiedergabe verbunden werden, das erfordert noch eine eigene Methodik des Bildes in der Edition, die hier nicht geleistet werden kann.“ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 183.

¹⁸⁶ vgl. dazu z.B. Hautzinger: „Neben dem Verlust eines feststehenden Textgefüges scheint [...] die Rückkehr zum Visuellen als eine der tiefgreifendsten Veränderungen unserer Schriftkultur durch die digitale Revolution“ Nina HAUTZINGER: Vom Buch zum Internet? Eine Analyse der Auswirkungen hypertextueller Strukturen auf Text und Literatur, St. Ingbert 1999, S. 25 oder auch Sahle: „Mit den bildlichen Repräsentationsformen und quellennahen Transkriptionsweisen gewinnt der unmittelbare visuelle Eindruck und der Blick auf die tatsächliche Überlieferung an Bedeutung. Die stark abstrahierenden, z.B. auf semantische Strukturen abzielenden Fragestellungen werden ergänzt, von einer historisierenden, materialistischen Wahrnehmung der Dokumente / Texte / Werke.“ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), 264f.

¹⁸⁷ Diese Überprüfbarkeit des Textes am Original forderte schon sehr früh Zeller. HANS ZELLER: Befund und Deutung. Interpretation und Dokumentation als Ziel und Methode der Edition, in: Texte und Varianten 1971, hrsg. v. Gunter MARTENS/Hans ZELLER, S. 45–90, hier S. 48–52.

werden in letzter Zeit vermehrt Techniken erprobt, wie diese Verknüpfung von Transkription und Bilddigitalisat technisch gelöst werden und wie sie anschließend in einer browserbasierten Umgebung am besten visualisiert werden kann. Wie später noch zu sehen sein wird, sind solche Methoden vor allem bei der Beschreibung textgenetischer Phänomene wichtige Hilfsmittel, um z.B. die Chronologie von Schreibprozessen zu analysieren. Auch in virtuellen Forschungsumgebungen können diese Methoden eine wichtige Rolle spielen, um die Diskussionen einer Textpassage direkt auf den betreffenden Bildausschnitt zu beziehen. Auf der Seite der Desktopanwendungen existieren schon seit längerem solche Annotationswerkzeuge. EPPT, Textgrid oder Digilib verfügen über ähnliche Möglichkeiten. Woran es aber bis heute mangelt, sind browserbasierte Lösungen, die das komplexere Annotieren von Bildausschnitten ermöglichen. In diesem Bereich zu nennen sind z.B. das UVic Image Markup Tool,¹⁸⁸ Tile¹⁸⁹ oder T-Pen¹⁹⁰. Diese Tools unterstützen vor allem zeilenbasierte Lösungsansätze, die einfache rechteckige Auszeichnungen zulassen und ein genaueres Annotieren in Form anderer geometrischer Formen nicht erlauben. Abschnitt 3.3 beschreibt die Implementierung eines solchen Werkzeuges und die Möglichkeiten wie damit textgenetische Prozesse analysiert, visualisiert und (ultra)diplomatische Ansichten erstellt werden können.

¹⁸⁸ http://www.tapor.uvic.ca/~mholmes/image_markup/index.php

¹⁸⁹ <http://mith.umd.edu/tile/>

¹⁹⁰ <http://t-pen.org/TPEN/>

2 Digitale Textgenetische Editionen

2.1 Exkurs: Textgenese in Druckeditionen

Im frühen 19ten Jahrhundert begann man erstmals, sich aus wissenschaftlichem Interesse mit historisch überlieferten Texten auseinanderzusetzen. Im Grunde reichen die Ursprünge aber noch weiter zurück bis in die Zeit der Renaissance des 15./16. Jahrhunderts. Das sich allmählich etablierende humanistische Weltbild, das wachsende Interesse an den Klassikern der antiken Literatur und nicht zuletzt die neuen technischen Möglichkeiten, die die Erfindung des Buchdrucks mit sich brachte, führten zu einer ersten (vor)wissenschaftlich geprägten, kritischen Auseinandersetzung mit dem Medium Text. Wenn auch noch nicht methodologisch fundiert, so doch schon in dem Sinne 'kritisch', dass aus einer Reihe von Textüberlieferungen die vermeintlich 'besten' Handschriften ausgewählt und sprachlich-stilistisch dem allgemeinen Empfinden der Zeit angepasst wurden. In Deutschland ist die Entstehung der Editorik als Wissenschaft eng mit dem Namen Karl Lachmann verbunden. Lachmann (1793-1851), Professor für lateinische und deutsche Philologie, etablierte eine normative Editionswissenschaft, deren Leistung darin bestand, Methoden der Textkritik mittelalterlicher und antiker Texte so zu formalisieren, dass sie auch für Texte neueren Ursprungs angewandt werden konnten. Er ging davon aus, Texte und Textvarianten unterschiedlicher Gattungen und unterschiedlicher Epochen mit den selben textkritischen Methoden, wie sie für mittelalterliche und antike Überlieferungen von Texten schon seit längerem verwendet wurden, so bearbeiten zu können, dass sich aus ihnen schliesslich ein nicht mehr existenter Urtext, der Archetyp, herauskristallisieren ließe.¹⁹¹ Der Editor handelte sozusagen als „Vollstrecker des Autorwillens.“¹⁹² Für die älteren Texte war diese Vorgehensweise auch durchaus sinnvoll. Die größtenteils verloren gegangenen Originaltexte konnten, wenn überhaupt, dann nur durch eine kritische Durchsicht und Bearbeitung der noch existierenden Abschriften (und deren Abschriften wiederum) rekonstruiert werden. Es galt also die durch die vielen Abschriften entstandenen Fehler zu lokalisieren und zu korrigieren.¹⁹³ Durch die Methoden der Recensio, der Emendatio und der Detectio sollte dieses Ziel erreicht werden.¹⁹⁴ Die Recensio - „Ermittlung aller relevanten Zeugen bzw. Träger eines Textes oder Werkes und die textkritische Prüfung ihrer Beziehungen zueinander“¹⁹⁵ stellt dabei die

¹⁹¹ Vgl. Bodo PLACHTA: Editionswissenschaft. Eine Einführung in Methode und Praxis der Edition neuerer Texte, 2. ergänzte und aktualisierte Auflage, Ditzingen 2006, S. 27ff.

¹⁹² Ebd., S. 91.

¹⁹³ Vgl. SAHLE: Digitale Editionsformen 1 (wie Anm. 1), S. 20.

¹⁹⁴ Vgl. Bruno MEYER: Zur Edition historischer Texte, in: Schweizerische Zeitschrift für Geschichte (SZG 1) 1951, S. 177-202, S. 180.

¹⁹⁵ PLACHTA: Editionswissenschaft (wie Anm. 191), S. 138.

erste Stufe des Editionsprozesses dar. In einem zweiten Schritt sorgte die Emendatio für die Bereinigung dieser Fehler. Dabei ist zu unterscheiden zwischen der Verbesserung von lediglich zufällig entstandenen Abschreibefehlern/Übertragungsfehlern, der sogenannten ‘emendatio ope codicum’ und einer ästhetischen Korrektur des Textes (coniectura). Die auch Konjekturealkritik genannte Methode machte es sich „zur Aufgabe, dem Autor nachfühlend und seinen Stil nachbildend [...] auch dort ‘Fehler’ zu beseitigen, wo sie selbst von den verfügbaren Handschriften nicht identifiziert wurden.“¹⁹⁶ Die Detectio Originis schliesslich versuchte vom „Wortlaut aus den Verfasser und die Entstehung des Werkes zu erforschen“.¹⁹⁷ Zeit seines Lebens wollte Lachmann seine Methode als universal anwendbar auf alle Textgattungen verstanden wissen. Bezeichnenderweise umfasst sein Lebenswerk neben Editionen über die Werke antiker Autoren wie Lucretius, Properz oder Catull auch Klassiker der mittelhochdeutschen Literatur wie Hartmann von der Aue, Iwein, Wolfram Eschenbachs Parzival oder das Nibelungenlied sowie eine Kritische Ausgabe von Lessings sämtlichen Werken. Schon damals erkannte man, dass die Konjekturealkritik der Texte, sowie die Fokussierung der Editionspraxis auf einen nur selten nachzuweisenden Urtext nicht ganz unproblematisch waren. So wurde im Laufe der weiteren Methodenentwicklung immer wieder heftig diskutiert, wie weit eine solche Textkritik gehen dürfe. Dem Verständnis der Textkritik als ästhetischem Kompositionsvorgang stand die Forderung nach Bewahrung der Originalität und Treue der Wiedergabe entgegen.¹⁹⁸ In den Geschichtswissenschaften wurden zu jener Zeit auch vermehrt Versuche unternommen, die historisch überlieferte Form beizubehalten und auf Konjekturealkritik gänzlich zu verzichten, um eine „größtmögliche Nähe zum Original“¹⁹⁹ zu gewährleisten. Nach Auffassung der Vertreter dieser Richtung bestand die Aufgabe des Editors vielmehr darin, die Originalität des Textes dokumentarisch festzuhalten und die Varianten ausschliesslich im Fussnotenapparat zu verzeichnen. Es muss allerdings angemerkt werden, dass sich die Vertreter dieser Leithandschriftenmethodik nicht wirklich durchsetzen konnten und grösstenteils die historisch kritische Methode Lachmanns angewendet wurde. So auch bei einem der grössten historischen Editionsprojekte, das in den 1830er Jahren seinen Anfang nahm, der MGH. Genau wie von Lachmann gefordert, stand bei diesem Projekt der Autor als literarisches Subjekt im Mittelpunkt, und so war das eigentliche Ziel der Editoren der MGH auch das Rekonstruieren eines vermeintlichen Urtextes. Das hier eine eher philologische Editionspraktik anscheinend unreflektiert übernommen wurde, verwundert ein wenig und ist wohl nicht zuletzt aus nationalpolitischen Motiven der damaligen Zeit zu erklären.²⁰⁰ Die ausgewählten Quellengattungen des Früh- und Hochmittelalters, die zur nationalen Identitätsbildung beitragen sollten, hatten häufig narrativen Charakter und konnten deshalb erfolgreich mit der philologisch mo-

¹⁹⁶ SAHLE: Digitale Editionsformen 1 (wie Anm. 1), S. 21.

¹⁹⁷ MEYER: Zur Edition (wie Anm. 194), S. 180.

¹⁹⁸ Vgl. SAHLE: Digitale Editionsformen 1 (wie Anm. 1), S. 42.

¹⁹⁹ Ebd., S. 44.

²⁰⁰ Vgl. Hartmut HOFFMANN: Die Edition in den Anfängen der Monumenta Germaniae Historica, in: Mittelalterliche Texte. Überlieferung – Befunde – Deutungen. Kolloquium der Zentralkommission der MGH am 28./29. Juni 1996, (MGH Schr. 42) 1996, hrsg. v. Rudolf SCHIEFFER, S. 189–232, S. 189.

tivierten Methodik Lachmanns ediert werden.²⁰¹ Die methodische Vorgehensweise, die sich über einen langen Zeitraum für die Reihe der MGH etablierte, wurde in der Geschichtswissenschaft zu einer Art Quasi-Standard, der auch heute noch in zahlreichen Editionsprojekten angewendet wird. Die historisch kritische Editionsmethodik in der Geschichtswissenschaft ist bis weit ins 20. Jahrhundert nicht mehr wesentlich weiterentwickelt worden.²⁰²

Anders verhält es sich in den Philologien. Impulse erhielt die Debatte um neue Methoden und Editionstechniken seit dem Beginn des 20. Jahrhunderts und vor allem nach 1945 fast ausschliesslich aus den Sprach- und Literaturwissenschaften. Dies ist begründet in einer Verschiebung der Interessensschwerpunkte, sowie in der „metatheoretische[n] Wandlung[...] von Textbegriff und Textverständnis.“²⁰³ In den Philologien etablierte sich das Editionswesen zu einer eigenen Wissenschaft.²⁰⁴ In den Geschichtswissenschaften hingegen ist die Editorik nie in den Kanon der klassischen Hilfswissenschaften aufgenommen worden. Die Entwicklung unterschiedlicher Editionsmodelle in den Philologien geht im Gegensatz zu den historischen Modellen eher in die Breite. Nach Sahle zeigt sich die philologische Editionswissenschaft „heute als eine pluralistische Methodenlehre, die versucht, sowohl hoch entwickelte Einzelmodelle, als auch viele konkurrierende Ansätze abzudecken.“²⁰⁵ So gewann z. B. eine Richtung in der Editionstheorie immer mehr an Bedeutung, die nicht mehr auf der Suche nach dem Urtext eines Autors war, sondern die bedingt durch die guten Überlieferungssituationen neuerer Texte den Autorwillen mehr und mehr in den Fokus rückte. Es fand eine Verschiebung von der Archetyp-Theorie hin zu einer Editionspraxis statt, die nun genau diametral die Ausgabe letzter Hand bevorzugte. In diesem Modell galt es die letzte vom Autor zu Lebzeiten autorisierte Fassung zu edieren. Der Herausgeber nahm sozusagen die Rolle als Testamentsvollstrecker des Autors an.²⁰⁶ Plachta verweist hier am Beispiel Goethes *Die Leiden des jungen Werthers* auf die Grenzen einer zu einseitigen Gewichtung der ersten oder der letzten Fassung.²⁰⁷ Autorisiert von Goethe waren sowohl die Fassung der Epoche des *Sturm und Drang* als auch die letzte Fassung der Weimarer Ausgabe, die gerade diese charakteristischen Sturm und Drang-Elemente nicht mehr enthielt. Je nachdem, auf welchen literarischen Kriterien der Editionsschwerpunkt liegt, ist es legitim, sowohl die erste als auch die letzte Fassung als Textgrundlage auszuwählen.

Aus solchen Problemen resultierte dann auch eine neue Entwicklung innerhalb der Editionspraxis. Anstatt eine von den verschiedenen Textfassungen zu bevorzugen, versuchte man nun den Fokus auf die den Text konstituierenden genetischen Prozesse zu lenken. Für die Vertreter dieses Editionsmodells war das eigentliche Werk erst aus dem

²⁰¹ Vgl. SAHLE: Digitale Editionsformen 1 (wie Anm. 1), S. 89.

²⁰² Vgl. ebd., S. 90.

²⁰³ Ebd., 107f.

²⁰⁴ An der Freien Universität Berlin wird beispielsweise der Masterstudiengang Editionswissenschaft als Interdisziplinärer Studiengang am Institut für Deutsche und Niederländische Philologie angeboten.

²⁰⁵ SAHLE: Digitale Editionsformen 1 (wie Anm. 1), S. 110.

²⁰⁶ Vgl. Ansgar NÜNNING: Ansätze - Personen - Grundbegriffe, in: Metzler Lexikon Literatur- und Kulturtheorie 2004, S. 133.

²⁰⁷ Vgl. PLACHTA: Editionswissenschaft (wie Anm. 191), S. 75ff.

Verlauf eines dynamischen Prozesses heraus zu verstehen. Diente die Erstellung eines Überlieferungs-Stemmas bei altphilologischen Texten noch der Rekonstruktion des Urtextes in Form einer chronologischen Darstellung überlieferter Textzeugen und vermeintlicher Archetypen, so ging es bei der Methodenentwicklung für vor allem neuere Texte um die Darstellung literarischer Produktion. Vor dem Hintergrund einer sich verändernden Vorstellung des Autor- und Textbegriffs ist hier besonders die Theorie der *critique génétique* zu erwähnen, die seit den 1960er Jahren vor allem von Autoren wie Foucault beeinflusst wurde. Hierzu Porombka: „Die critique génétique hat sich vorgenommen, [...] Netzwerke und vor allem ihre Dynamik sichtbar zu machen. Und das geht am besten dort, wo der Text noch nicht seine scheinhafte Endgültigkeit im Druck erlangt hat, sondern sich in flüchtigen Aufzeichnungen objektiviert. Folgerichtig wird statt des Textes auf die Textualität geschaut, und an die Stelle des Produktes tritt die emphatische Auseinandersetzung mit der Produktion“²⁰⁸ und weiter: „Die konsequentesten Interpretationen der critique génétique gehen davon aus, dass die Skizzen, Entwürfe, Notate und Varianten gerade in ihrer Flüchtigkeit und Vorläufigkeit als eigenständige Texte mit eigenständiger Ästhetik gelten müssen. Und sie gelten zugleich als Teilstücke eines großen Textes, der alle Produktionsstufen umfasst. Dieser große Text setzt sich zusammen aus der Summe der 'Texte' der jeweils vorhandenen, historisch genau fixierbaren Textfassungen, die zu dem Werk überliefert sind; sie insgesamt bilden das, was im editorischen Sinne als 'Text' eines Werkes zu bezeichnen ist.“²⁰⁹ Anders etwa als bei der Editionswissenschaft deutscher Tradition, hat sich bis heute zwar noch kein eigenständiges theoretisches Textgenese-Modell (in Form von textkritischen Apparaten etwa) aus ihr entwickelt,²¹⁰ dennoch ist sie eine der viel zitiertesten und oft auch umstrittensten Theorien neuerer Zeit und ein wichtiger Impulsgeber in der wissenschaftlichen Analyse dynamischer Textprozesse. Das Hauptziel der critique génétique ist es, „durch eine grundsätzliche Analyse der schriftlichen Spuren den Prozeß der textlichen Entstehung zu rekonstruieren und daraus wo möglich generelle Modelle oder wenigstens Typologien für das literarische Schaffen zu entwickeln.“²¹¹ Die Entwicklung der Editionswissenschaft in Frankreich hat sich schon relativ früh von der deutschen stark philologisch geprägten getrennt. Nach dem verlorenen Deutsch-Französischen Krieg von 1870/71 wurde sie als „Wissenschaft der Sieger“²¹² eher gemieden, und es bildete sich keine vergleichbare Editionstheorie heraus. Die französische Editionswissenschaft hatte es sich in erster Linie zum Ziel gesetzt, „einem breiten Publikum zuverlässige Texte zu liefern.“²¹³ Die komplette Verzeichnung aller Textzeugen und Varianten in textkritischen Apparaten war eher unüblich.

²⁰⁸ Stephan POROMBKA: Literaturbetriebskunde. Zur 'genetischen Kritik' kollektiver Kreativität, in: Kollektive Kreativität (Jahrbuch für Kulturwissenschaften und ästhetische Praxis) 2006, hrsg. v. Stephan POROMBKA/Wolfgang SCHNEIDER/Volker WORTMANN, S. 72–87, S. 77.

²⁰⁹ Ebd., S. 77.

²¹⁰ Almuth GRÉSILLON/Frauke ROTHER: Literarische Handschriften. Einführung in die 'critique génétique', Bd. 4 (Arbeiten zur Editionswissenschaft), Bern u.a. 1998, S. 9.

²¹¹ Jean-Louis LEBRAVE: Genetische Textkritik und Edition in Frankreich. Beiträge zur VII. Internationalen Fachtagung der Arbeitsgemeinschaft philosophischer Editionen (12.-14. März München), in: Editio. Beihefte, Teil 11 1998, S. 113–122, S. 119.

²¹² Ebd., S. 113.

²¹³ Ebd., S. 113.

Die Anfänge der critique génétique reichen zurück bis in die 1960er Jahre.²¹⁴ Das Projekt, das sich der Erforschung der von der Bibliothèque Nationale erworbenen Handschriften Heinrich Heines widmete, bildete den Ausgangspunkt dieser Entwicklung.²¹⁵ Die Erforschung der Heine-Handschriften in diesem Projekt diente nicht dem Zweck, eine Art textkritische Edition anzufertigen, die auf den Rezipienten einer solchen Edition, also den Leser, fokussiert war. Das Interesse der Forscher richtete sich vielmehr auf den Autor Heine selbst und seine Schreibprozesse.²¹⁶ Besonders die Arbeitsmanuskripte Heines rückten die Textgenese ins Zentrum: „Die Arbeitsmanuskripte schienen doch den neu aufgebauten Begriffen der textuellen Produktivität viel angemessener zu sein als die Texte [selbst]“²¹⁷ Es ging nicht, wie in der deutschen Editionstradition um die „Herstellung der 'richtigen' Textgestalt“²¹⁸, sondern um den Prozess der kreativen Textproduktion, und wie der Prozess des Schreibens wiederum Einfluss nahm auf das Schreiben selbst: „die Schrift [ist] nicht wie für das Buch als Zeugnis eines geregelten Vervielfältigungsverfahrens, sondern als Spur eines Schreibprozesses zu verstehen. [...] So wird auch die 'dritte Dimension des Textes' sichtbar, nämlich die der zeitlichen Textentstehung.“²¹⁹ Ein zentraler Gedanke der deutschen Editionswissenschaft wurde dabei in Frage gestellt. Die Trennung von Befund und Deutung hielt man (zumindest für diese Art der Textüberlieferung) nicht für durchführbar: „..., daß die saubere Trennung von Befund und Deutung, so wie sie von Editoren erwünscht wird, praktisch nur schwer realisierbar ist, es sei denn, die semiotische Deutung des graphischen Befunds werde in den Befund selbst als selbstverständlich interpretiert...“²²⁰, es sich also um keinen zielbewusst entwickelten Text handelte, wie ihn etwa Beißner in seinem Treppenapparatmodell voraussetzte (siehe Abschnitt 2.1.1 Treppen- oder Stufenapparat). Tendierte man in Deutschland also mehr zur Verzeichnung dynamischer Prozesse in genetischen Apparaten, so „ist die critique génétique einen anderen Weg gegangen: Sie hat behauptet, der Entstehungsprozess sei ebenso interessant wie das Ergebnis des Schreibens, und er verlange nach spezifischen Konzepten und Untersuchungsmitteln, die - wenigstens in einer ersten Phase - eine Abkehr von den überlieferten Modellen erfordere, also eine Abkehr vom Textbegriff, vom Variantenapparat und von allen der Philologie zugehörigen Begriffen.“²²¹ Aufgrund des großen Interesses, das diesen Fragestellungen in der französischen Literaturwissenschaft entgegengebracht wurde, kam es 1982 zur Gründung des *Institut des textes et manuscrits modernes* (I.T.E.M.).²²² Die 'généticiens', wie sich die Wissenschaftler dieser Fachrichtung auch nennen, sehen

²¹⁴ Geert LERNOUT: 'Critique génétique' und Philologie, hrsg. v. Rüdiger NUTT-KOFOTH, 2000, S. 121–142, S. 121.

²¹⁵ Ebd., S. 27.

²¹⁶ Vgl. Klaus HURLEBUSCH: Den Autor besser verstehen: aus seiner Arbeitsweise. Prolegomenon zu einer Hermeneutik textgenetischen Schreibens, in: Textgenetische Edition 1998, hrsg. v. Hans ZELLER, S. 7–51, S. 28.

²¹⁷ LEBRAVE: Genetische Textkritik (wie Anm. 211), S. 114.

²¹⁸ Almuth GRÉSILLON: Bemerkungen zur französischen 'édition génétique', in: Editio. Beihefte, Teil 10 1998, S. 52–64, S. 51.

²¹⁹ LEBRAVE: Genetische Textkritik (wie Anm. 211), S. 114 vgl. dazu auch Louis Hay: Critiques du manuscrit. In Hay (Hrsg.): La naissance du texte. Paris 1989, S. 10.

²²⁰ Ebd., S. 117.

²²¹ Ebd., S. 118.

²²² LERNOUT: Critique génétique (wie Anm. 214), S. 121.

sich in der Tradition der französischen strukturalistischen und poststrukturalistischen Literaturwissenschaft verankert, die wenig mit der historischen Forschung zu tun hatte.²²³ Ziel der *critique génétique* ist nicht „die Interpretation vollendeter Texte“²²⁴, bei ihr wird eine Varianz „nicht mehr teleologisch als Vorstufe zur letzten Fassung [...] betrachtet.“²²⁵ Sie wertet vielmehr das Schreiben an sich auf. Das Schreiben „löst sich aus dem Dienstverhältnis zum Werk, wenn diese im Prinzip nur als Durchgangsstadium des Schreibprozesses verstanden wird.“²²⁶ *Généticiens* verstehen das „Schreiben als 'Praxis' [...] d.h. als Tätigkeit, deren Sinn nicht außerhalb ihrer selbst liegt und die deshalb höherrangig ist als das, was aus ihr hervorgeht.“²²⁷ Der Anspruch an eine nach der Philosophie der *critique génétique* hergestellten Edition ist sehr hoch, es geht dabei um nichts weniger als „die lückenlose Darstellung aller handschriftlichen (bzw. maschinenschriftlichen) Zeugen, die an einer Textgenese teilhaben, und zwar so, daß chronologisch alle Zeugen nacheinander, Blatt für Blatt, in diplomatischer oder linearisierter Umschrift, und wenn möglich, unter Beigabe der entsprechenden Faksimiles dargestellt werden.“²²⁸ Das Augenmerk wird dabei vor allem auf die Raum-Zeit-Komponente textgenetischer Prozesse gelegt. Ein großes Problem bei der Verzeichnung dieser Prozesse in klassischen Druckeditionen liegt darin begründet, dass das zweidimensionale Medium Buch nicht geeignet ist, die dynamische Komponente (also eine weitere dritte Dimension) adäquat darzustellen. Die unterschiedlichen Apparatmodelle textkritischer Editionen erlauben zwar eine mehr oder weniger zufriedenstellende Kodierung der materiellen Befunde. Ihre Visualisierungsfähigkeit genetischer Prozesse ist jedoch stark begrenzt. Sowohl die topographische Verortung der Schreibprozesse, als vor allem auch die zeitliche Dimension (welche Bearbeitungsschritte und Veränderungen folgen in welcher Reihenfolge aufeinander), finden in den klassischen Apparatmodellen nicht den ihnen gebührenden Platz. Grésillon charakterisiert den Prozess der Handschriftenuntersuchung folgendermaßen: „Die Handschrift konfrontiert uns auf ihrem zweidimensionalen Raum mit festgefrorenen Schriftzeichen. Diese räumlichen Indizien werden dank positionellen, graphischen und sprachlichen Kriterien sozusagen in zeitlich-genetische Indizien uminterpretiert; d. h. der Forscher bereichert den zweidimensionalen Raum um die zeitliche Dimension der Schreibprozesse, die es ihm gestattet, einen Handschriftenfund genetisch zu interpretieren.“²²⁹ Und Hurlebusch stellt hierzu fest: „Der visuelle Akt des Schreibens vollzieht sich räumlich-zweidimensional, der unsichtbare des Lesens in linearer Eindimensionalität. Die spatiale Lektüre eines korrigierten handschriftlichen Textes ist nicht mit der linearen Buchtextlektüre zu harmonisieren.“²³⁰ Auf die Rolle, die dabei den Digitalen Editionen zukommen könnte, verweist schon Grésillon: „Erst die komplette Repräsen-

²²³ LERNOUT: *Critique génétique* (wie Anm. 214), S. 122ff.

²²⁴ Ebd., S. 125.

²²⁵ SAHLE: *Digitale Editionsformen 2* (wie Anm. 2), S. 276.

²²⁶ HURLEBUSCH: *Hermeneutik* (wie Anm. 216), S. 11.

²²⁷ Ebd., S. 15.

²²⁸ GRÉSILLON: *édition génétique* (wie Anm. 218), S. 54.

²²⁹ Ebd., S. 54.

²³⁰ auch wenn Hurlebusch hier von der Eindimensionalität des Lesens spricht und den Akt des Schreibens somit als zweite und nicht als dritte Dimension kennzeichnet, so ist doch das gleiche gemeint, wie bei Grésillon. HURLEBUSCH: *Hermeneutik* (wie Anm. 216), S. 23.

tation der Textgenese, so wie sie Hypertext-Computerprogramme gestatten, mit der Möglichkeit, jede Textstufe beliebig auf dem Bildschirm abzurufen, sie in Verbindung zu setzen mit jeder beliebigen anderen Textstufe, mit der Leichtigkeit, genetische Kommentare mit einzugeben oder genetische Editionen je nach Bedarf auszudrucken ...“ würde der Forderung nach der Entwicklung der Textgenetik hin zu einem eigenen neuen Wissenschaftszweig „... zum endgültigen Durchbruch verhelfen.“²³¹

Während in den (neueren) Literaturwissenschaften das Interesse an genetischen Prozessen vor allem von dem Wunsch getragen ist, kreative Schreibprozess sichtbar zu machen, war man in der Mediävistik lange Zeit auf der Suche nach der ‘besten’ Textüberlieferung, wobei beste hier unterschiedliches meinen kann, etwa die Ursprungsgestalt des Textes, die z.B. bei der Bibelexegese eine wichtige Rolle spielte. Die Textgenese kann dabei häufig schon vor dem eigentlichen Prozess der schriftlichen Fixierung einsetzen. So beschreibt Bein ausführlich den Prozess mittelalterlicher Textproduktion.²³² Er gibt zu bedenken, dass mittelalterliche Handschriften immer nur „fixe Momentaufnahmen einer Textgeschichte“²³³ sind und dass der „Blick auf den ‘Urheber’, den Autor des Textes und auf das, was dieser Urheber einstmals in die Welt entlassen hat, mehr als getrübt ist.“²³⁴ Schriftlich fixierte Texte haben in der Regel eine Vorgeschichte, die gekennzeichnet ist von einem häufigen Wechsel zwischen mündlicher und schriftlicher Überlieferung. Literarische Texte des Mittelalters etwa sind häufig für Aufführungszwecke bestimmt gewesen.²³⁵ Sie haben sich in einem mehrfachen „Wechsel des Aggregatzustandes“²³⁶ befunden: „... einmal finden sie sich fixiert auf Wachs oder Pergament, dann wieder im ephemere (flüchtigen, nicht fixierten) Zustand der Oralität (Mündlichkeit).“²³⁷ Dieser Zustand kann durchaus auch für einen einzelnen Textzeugen mehrfach wechseln. Betont werden bei Bein auch ausführlich die unterschiedlichen Rollen der Personen die an solch einer Art der Textproduktion beteiligt waren (Auftraggeber, Mäzene, Autoren, Redaktoren, Schreiber, Textinterpreten).²³⁸

Über die Mediävistik und die Literaturwissenschaft hinaus gibt es aber auch Gründe, die für eine größere Beachtung textgenetischer Prozesse etwa in der (neueren) Geschichtswissenschaft sprechen. Das folgende Beispiel soll dies illustrieren. Es stammt aus einem Editionsprojekt, das der Autor dieser Arbeit betreut und welches sich zur Aufgabe gesetzt hat, alle Nuntiaturreporte Eugenio Pacellis, des späteren Papstes Pius XII., die während seiner Zeit als Apostolischer Nuntius in München und Berlin entstanden sind, in einer Online-Edition zur Verfügung zu stellen. Auf Beispiele aus diesem Projekt wird im Verlaufe der vorliegenden Arbeit noch häufiger zurückgegriffen werden. Aus Pacellis Zeit als päpstlicher Nuntius in München und Berlin

²³¹ GRÉSILLON: édition génétique (wie Anm. 218), S. 62.

²³² Thomas BEIN: Textkritik. Eine Einführung in Grundlagen germanistisch-mediävistischer Editionswissenschaft, Frankfurt am Main (u.a.) 2007.

²³³ Ebd., S. 20.

²³⁴ Ebd.

²³⁵ Ebd., S. 23.

²³⁶ Ebd.

²³⁷ Ebd.

²³⁸ S. 22 ebd., Die Verflechtungen der einzelnen Personen und ihr Einwirken auf den Textentstehungsprozess eines Werkes wird bei Bein sehr anschaulich in einem Textmodell dargestellt.

haben sich sowohl die Ausfertigungen seiner Berichte und die Weisungen im Päpstlichen Staatssekretariat als auch die Entwürfe in den Nuntiaturarchiven München und Berlin erhalten. Da Pacelli fast täglich nach Rom schrieb, oft auch mehrmals am Tag, haben sich ca. 7000 Nuntiaturberichte erhalten, die bis zu 50 Seiten umfassen können. Hinzu kommen noch ca. 7000 Entwürfe der Berichte, 6000 Weisungen des Staatssekretariats und ca. 5000-6000 Anlagen unterschiedlichster Art, wie Denkschriften, Briefe, Zeitungsartikel etc. In einem Nuntiaturbericht geht es unter anderem um die Charakterisierung der Führungsspitze der kommunistischen Münchener Räterepublik nach einem Treffen Lorenzo Schioppas, Auditor der Nuntiatur, mit Max Levien, dem Führer der Räterepublik. Der Nuntiaturbericht ist schon länger bekannt und dient bis heute als Beweis für den auch bei Pacelli (vermeintlich) existenten Antisemitismus. Die allgemeine Praxis Pacellis, solche Nuntiaturberichte zu verfassen, bestand in der Regel in einem mehrstufigen Entwurfsprozess, an dem verschiedene Personen beteiligt sein konnten (Stenotypisten, Sekretäre und er selbst). Die hervorgehobene Textstelle

Schreiber: Stenotypist Schioppa Pacelli
Betreff: L'antemitismo della <La> Nunziatura e la <R>epubblica dei Consigli

Come ebbi l'onore di riferire a V. E. R. col mio cifrato N. [sic] <poco dopo inaugurata la Repubblica dei Consigli> al principio della corrente settimana> due Legazioni estere in Monaco furono invase dalla guardia rossa della Repubblica dei Consigli. Inizialmente <In seguito, venne richiesto> alla Legazione di Prussia <è stato> <venne> <richiesto> l'automobile, ed è stato <perfino arbitrariamente> arrestato il Console generale Austro-Ungarico e non rilasciato se non in <seguito> <dietro> <alle> energiche proteste dell'incaricato d'Affari d'Austria-Ungheria.

In seguito a tali deplorevoli avvenimenti <è stato> <si> è creduto opportuno <ind-etta> <circa> <di indire> una riunione del Corpo diplomatico per deliberare in proposito. Dopo una lunga discussione è stato deciso di parlare della cosa direttamente con Levien, capo della Repubblica dei Consigli di Monaco. <, e di> <per> costringerlo a dichiarare apertamente <senza equivoci> se e come l'attuale Governo comunista intenda riconoscere e tutelare le immunità delle Rappresentanze diplomatiche. Della cosa sono stati incaricati <Le trattative vennero assegnate> <affidate a> la Nunziatura e <d al> la Legazione di Prussia. Siccome <a giudizio di tutti> sarebbe stato assolutamente indecoroso per

Il Levien è un giovanotto, anche egli russo ed ebreo, di circa trenta o trentacinque anni. Pallido, sporco, dagli occhi scialbi, <dalla voce rauca e sguaiata> un vero tipo ributtante, eppure con una fisionomia intelligente e furba.

palazzo <già reale dei> Wittelsbach, <già> <residenza del Re> <di> <Baviera>. Lo spettacolo, che <ora> presenta detto palazzo, è indescrivibile. La confusione più caotica, il sudiciume più nauseante, l'andirivieni <continuo> di soldati <ed> <e di> operai armati,

154v
le grida, le parole sconcie, le bestemmie, che ivi risuonano, rend <a> <ono> <la> <quella>, che fu la residenza> prediletta <casa> del Re <gnante> di Baviera, una vera bolgia infernale. Un esercito di impiegati, che vanno, che vengono, che trasmettono ordini, che propagano notizie, e fra essi una schiera di giovani donne, dall'aspetto poco rassicurante, ebreo come i primi, che stanno in tutti gli uffici, con arie provocanti e con il <orris> <a> <equivoci>. A capo di questo gruppo femminile vi è l'amante di Levien: una giovane russa, ebrea, divorziata, che comanda da padrona. Ed a costei <si> <è> <la> Nunziatura ha> dovuto <far di> <appello> <pur troppo> inchinarsi> per avere il biglietto di libero passaggio <una> <vera> <indegnità>

Il Levien è un giovanotto, anche egli russo ed ebreo, di circa trenta o trentacinque anni. Pallido, sporco, dagli occhi scialbi, <dalla voce rauca e sguaiata> un vero tipo ributtante, eppure con una fisionomia intelligente e furba. Appena <è> <S> è degnato <appena> di ricevere <Mons.> Uditore in un corridoio, <scortato> <circondato> da una guardia <scorta> armata, fra cui un gobbo anche egli armato, che è la sua guardia fedele. Col cappello in testa e fumando, ha ascoltato quan <è> <ct> o Mor. Schioppa gli esoneva. protestando rietutamente <e> <scarbatamente> che aveva fretta

Abbildung 2.1 Textgenese am Beispiel eines Nuntiaturberichts Eugenio Pacellis

lautet in der Übersetzung etwa folgendermaßen:

„Levien ist ein junger Mann, auch er Russe und Jude, von circa 30 oder 35 Jahren. Bleich, schmutzig, mit ausdruckslosen Augen <mit einer kratzigen und groben Stimme:> ein wirklich widerlicher Typ, nichtsdestotrotz mit einer intelligenten und schlaun Physionomie.“

Der Vorwurf des Antisemitismus kann hier nun insofern präzisiert werden, als sich anhand einer textgenetischen Analyse des Entwurfs zeigen lässt, wie die stereotypen

antisemitischen Klischees Eingang in diesen Bericht gefunden haben. Der Text stammt ursprünglich nicht von Pacelli selbst, sondern wurde von einem Stenotypisten aufgezeichnet (blaue Schrift), sehr wahrscheinlich von Schioppa diktiert (da nur dieser und nicht Pacelli selbst bei dem Treffen anwesend war). In einem zweiten Bearbeitungsvorgang (grüne Schrift) wurden dann Ergänzungen von Schioppa hinzugefügt. Pacelli selbst nahm in einem letzten Bearbeitungsschritt abschließende Änderungen vor, allerdings ohne besagte Textstelle zu korrigieren. Mit seiner Unterschrift autorisierte Pacelli schließlich den Bericht und machte sich somit die negative Charakterisierung Leviens zu eigen. Von allen Entwürfen wurde in einem letzten Verwaltungsakt eine Abschrift angefertigt, die als Ausfertigung an den Heiligen Stuhl gesandt wurde. Alleine aus der Ausfertigung ist also nicht zu erkennen, wie der Bericht entstanden ist und wer an der Abfassung des Textes beteiligt war. Eine historische Bewertung dieses Befundes kann hier nicht geleistet werden. Gerade ein solches Beispiel zeigt jedoch, wie wichtig die exakte Analyse solcher Prozesse sein kann, um zu einer wissenschaftlich fundierten Beurteilung eines Sachverhaltes zu gelangen.

2.1.1 Apparate

Eine zentrale Rolle in Bezug auf die Analyse und Darstellung textgenetischer Prozesse in der Deutschen Literaturwissenschaft spielt seit der Herausbildung der modernen Editionswissenschaft im 19. Jahrhundert der kritische Apparat. Dieser dient dem Verständnis des Werkes selbst, so wie seiner Entstehungs- und Überlieferungsgeschichte. Komplementär zum konstituierten Text enthält der Apparat alle Informationen, die nicht in ihm selbst verzeichnet sind. Hierbei kann es sich um Texterläuterungen in Form von kritischen Kommentaren, Wort- und Sacherklärungen, Anmerkungen, Literaturhinweisen, intertextuellen Bezugstexten, etc. handeln.²³⁹ In den Anfangszeiten der modernen Editionswissenschaft diente der Apparat in erster Linie als Legitimation für die Erstellungs- und Auswahlkriterien des konstituierten Textes. „Die Varianten wurden [...] in ihrer Beziehung zum edierten Text, nicht in ihrer Beziehung zueinander verzeichnet.“²⁴⁰ Mit dem Wandel von der kritischen hin zu einer dokumentarischen Editionswissenschaft verschob sich jedoch der Schwerpunkt der editorischen Tätigkeit. Stand vormals noch die Konstitution eines autorisierten Textes im Zentrum der Editionswissenschaft, so rückte im Laufe der Zeit immer mehr die Analyse und Darstellung seiner unterschiedlichen Fassungen in den Mittelpunkt.²⁴¹

Die immense wissenschaftliche Bedeutung, die der dokumentarische Apparat in den Editionswissenschaften erlangte, lässt sich daran erkennen, wie der eigentliche konstituierte Text zugunsten des Apparates immer mehr in den Hintergrund trat. In diesem Sinne stellte der Editionstext nur noch eine komprimierte selektive Darstellung der

²³⁹ http://www.uni-saarland.de/fak4/fr41/germanistik/pdfs/studieninfos/infos_neueStO/Info_Basis_wissen_LitWiss.pdf

²⁴⁰ Hermann ZWERSCHINA: Variantenverzeichnung, Arbeitsweise des Autors und Darstellung der Textgenese, in: Text und Edition. Positionen und Perspektiven 2000, hrsg. v. Rüdiger NUTT-KOFOTH u. a., S. 203–231, S. 204.

²⁴¹ Anne BOHNENKAMP: Neugermanistische Editionswissenschaft, in: Kompendium der Editionswissenschaften, URL: http://www.edkomp.uni-muenchen.de/CD1/frame_edkomp_AB.html.

Textgeschichte dar, die erst durch den Apparat ihre wissenschaftliche Legitimation erfuhr. Maas und Boydston etwa verweisen hier auf die Hilfsfunktion des Textes zum besseren Verständnis eines komplexen Verweisungsgefüges, das erst durch den Apparat sichtbar wird.²⁴² Insofern leisteten solche Apparate einen wichtigen Beitrag, um das Werk einer kritischen Rezeption und Diskussion zugänglich zu machen. Die klassischen kritischen Apparate in der Lachmannschen Tradition waren aber nur bedingt dazu geeignet, die neuen dokumentarischen Anforderungen zu erfüllen und die Textgenese adäquat darzustellen. Auf die Problematik der Verzeichnung von Variantenüberlieferungen in herkömmlichen Apparaten Lachmannscher Prägung wiesen schon Seuffert²⁴³ und Backmann²⁴⁴ zu Beginn des 20. Jahrhunderts hin. Das Hauptproblem sahen sie in der traditionellen Verzeichnung der varianten Textstellen, ohne näher auf ihren Entstehungszusammenhang einzugehen. Zusammenhänge von Veränderungen an räumlich getrennten Textstellen sollten die Textgenese sichtbar machen. Backmann spricht hier auch davon, die 'absolute Chronologie' von Veränderungen (vertikale oder paradigmatische Ebene) durch die 'relative Chronologie' (horizontale oder syntagmatische Ebene) miteinander in Beziehung zu setzen.²⁴⁵ Für ihn gibt „die Klarlegung der Entwicklung [...] dem Apparat erst seinen selbständigen Wert gegenüber dem Textabdruck, ja sie gibt ihm, wenn sie in der rechten Weise erfolgt, ein Übergewicht an Bedeutung über den letzteren.“ Auch Zwerschina sieht in dieser Art der Apparatgestaltung den „entscheidenden Fortschritt gegenüber den altphilologischen Variantenverzeichnungen.“²⁴⁶ Allerdings geht Backmann dann doch nicht soweit, den Entstehungsprozess als die eigentliche Leistung des Autors zu begreifen. Nach wie vor schätzte er den „edierten Schlußtext als fertig gestelltes Kunstwerk doch höher ein [...] als die Vor- und Zwischenstufen.“²⁴⁷

Dennoch hatte dieses neue Paradigma einen großen Einfluss auf die Editionswissenschaften. Im Laufe des 20. Jahrhunderts entwickelte sich so ein weites Spektrum an Apparatmodellen. Diese versuchten, möglichst genau, die Entstehungs- und Überlieferungsgeschichte der jeweiligen Textzeugen abzubilden. Dabei lassen sich die Apparattypen im wesentlichen zwei Hauptkategorien zuordnen: den lemmatisierten, die listenartig, meist in Form von Fußnotenapparaten, dem edierten Text angehängt werden²⁴⁸ und den integralen Apparaten, „die eine Darstellung des gesamten Textes einschließlich seiner Varianten - also eine gemeinsame Darstellung von variantem und invariantem Text - zum Ziel haben.“²⁴⁹ Hier gilt es wieder zu unterscheiden zwischen

²⁴² Vgl. Gunter MARTENS: Texterschließung durch Edition. Überlegungen zur rezeptionsästhetischen Bedeutung textgenetischer Apparate, in: LiLi (19.20) 1975, S. 82–104; sowie Jo Ann BOYDSTON: In Praise of Apparatus, in: Text 5 1991, S. 1–13, S. 9.

²⁴³ Vgl. Bernhard SEUFFERT: Prolegomena zu einer Wieland-Ausgabe. III. IV. Berlin (= Abhandlungen der preußischen Akademie der Wissenschaften), 1905 (Reprint Hildesheim 1989), S. 57.

²⁴⁴ Vgl. Reinhold BACKMANN: Die Gestaltung des Apparates in den kritischen Ausgaben neuerer deutscher Dichter, in: Euphorion 25 1924, S. 629–662, S. 638.

²⁴⁵ Ebd., S. 638.

²⁴⁶ ZWERSCHINA: Variantenverzeichnung (wie Anm. 240), S. 206.

²⁴⁷ Ebd., S. 206.

²⁴⁸ Dies kann entweder direkt im Anschluss eines Textabschnittes auf der selben Buchseite sein oder auch als Anhangapparat ganz am Ende des Textes.

²⁴⁹ BOHNENKAMP: Editionswissenschaft (wie Anm. 241). Hierzu vgl. auch Hans ZELLER: Die Ty-

den linearen Apparatypen, die die varianten Textstellen unmittelbar hinter den invarianten verzeichnen, sie sozusagen direkt einblenden (Einblendungsapparate) und solche, die möglichst genau dem Backmannschen Paradigma folgend, die paradigmatische Varianz in vertikaler Ebene und die syntagmatische in horizontaler Ebene betonen (Stufen- oder Treppenapparate).

Eine andere Einteilungsmöglichkeit besteht in der Unterscheidung zwischen der sogenannten Binnenvarianz und der Außenvarianz.²⁵⁰ Apparate, die die Varianz eines einzigen Textzeugen (Binnenvarianz) beschreiben, nennt man auch Einzelapparate. Die Außenvarianz verzeichnet Varianten, die in mehreren Textzeugen überliefert sind. Für diese Art der Textüberlieferung werden häufig synoptische Methoden oder Paralleldrucke verwendet. Auf die große Bedeutung der äußeren Gestaltung der unterschiedlichen Apparatypen weist Urchueguía hin: „Aufgrund ihrer komplexen inneren Struktur bilden A.e ein hierarchisches und normatives System der Informationsverzeichnung, -klassifikation und -verwaltung. Die Ebene der (typo)graphischen Gestaltung bildet dabei nicht nur eine Form der Vermittlung dieses Ordnungssystems, sondern sie hat selbst entscheidenden Anteil an der durch die Edition vermittelten Information.“²⁵¹ Aufgrund gattungsspezifischer Unterschiede, heterogener Überlieferungssituationen, immer spezialisierterer Fragestellungen an die Textentstehung, unterschiedlicher autorspezifischer Arbeitsweisen²⁵² und der Verwendung des Apparates als historisch bedingter Kulturtechnik²⁵³ haben sich so die unterschiedlichsten Apparatypen herausgebildet, die auch in Zwischenformen auftreten können. Im Folgenden sollen die wichtigsten von ihnen kurz beschrieben werden:

Einzelstellenapparat Der lemmatisierte Einzelstellenapparat stellt die klassischste Apparatform dar. Als Lemma wird hier der variierende Textteil bezeichnet, zu dem überlieferte Varianten existieren. Das Lemma wird in der Regel komplett im Apparat wiederholt, um so eine Zuordnung der Varianten zu ermöglichen. Die einzelnen Varianten werden durch das sogenannte Lemmazeichen vom Lemma getrennt. Als gängiges Symbol hat sich hierfür die nach links geöffnete eckige Klammer etabliert. Um die einzelnen Entstehungs- und Überlieferungsvarianten zu kennzeichnen, werden unterschiedliche Siglensysteme verwendet. Für mittelalterliche Texte (überwiegend handelt es sich hier um Überlieferungsvarianten) hat sich dafür eine Notation in Form von Abkürzungen des Fundortes des Textzeugen durchgesetzt (also etwa H für Heidelberger Handschrift). In der neueren Literaturwissenschaft bezeichnen die Siglen in der Regel die unterschiedlichen Arten der Überlieferungsträger.²⁵⁴ H: eigenhändige Handschrift, h: fremde Handschrift, J: Druck (bzw. erschienen) in einem regelmäßig erscheinenden

pen des Germanistischen Variantenapparats und ein Vorschlag zu einem Apparat für Prosa, in: Editionsprobleme der Literaturwissenschaft 105, Sonderheft (1986), S. 42–69.

²⁵⁰ BOHNENKAMP: Editionswissenschaft (wie Anm. 241).

²⁵¹ Cristina URCHUEGUIA: Apparat, in: Kompendium der Editionswissenschaften, URL: http://www.edkomp.uni-muenchen.de/CD1/frame_edkomp_CU0.html.

²⁵² ZWERSCHINA: Variantenverzeichnung (wie Anm. 240), S. 203.

²⁵³ URCHUEGUIA: Apparat (wie Anm. 251).

²⁵⁴ vgl. http://www.uni-saarland.de/fak4/fr41/germanistik/pdfs/studieninfos/infos_neueStO/Info_Ba_siswissen_LitWiss.pdf (zuletzt abgerufen 24.4.2012)

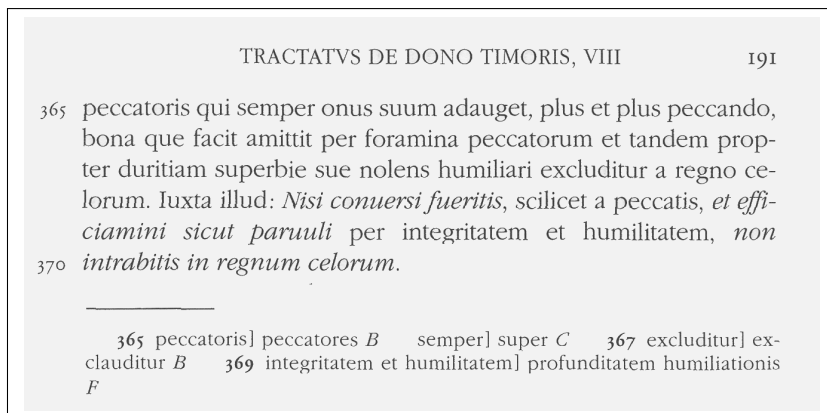


Abbildung 2.2 Lemmatisierter (positiver) Apparat.

Publikationsorgan, D: Druck (aktiv/passiv) autorisierte Drucke, Doppeldrucke, Raubdrucke, Titelaufgaben, S: Abdruck des Textes innerhalb der (gesammelten) Schriften des Autors. Neben dem lemmatisierten Apparat (positiver Apparat - Abbildung 2.2²⁵⁵) existiert auch der sogenannte nichtlemmatisierte Apparat (negativer Apparat - Abbildung 2.3). Dieser verzichtet im Gegensatz zum positiven Apparat auf die Zuordnung der Textvariante durch das Lemmazeichen und kann zusätzlich auch die invarianten Textstellen und Textträger verzeichnen. Der Vorteil des lemmatisierten Apparates liegt im Erfassen der Lesarten auf einen Blick.²⁵⁶

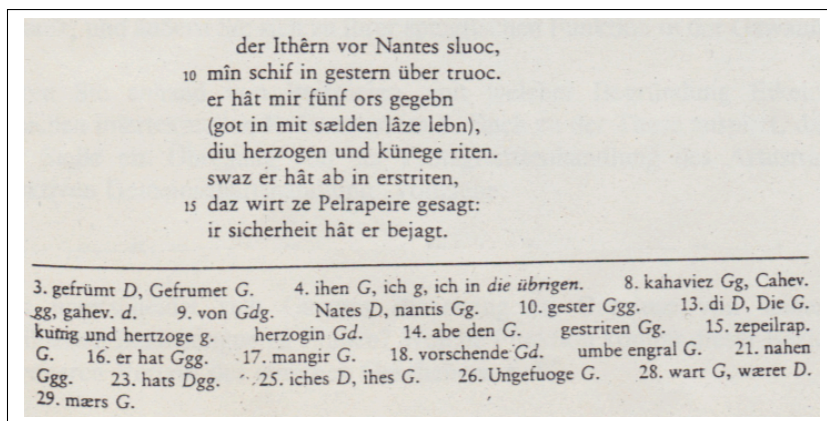


Abbildung 2.3 Nichtlemmatisierter (negativer) Apparat

²⁵⁵ Abbildung nach: de Romanis HUBERTUS: Exempla medii aevi, hrsg. v. Christine [Hrsg.] BOYER, Bd. 4: Hvmberti de Romanis De dono timoris (Corpus Christianorum : Continuatio mediaevalis; 218), Turnhout 2008, S. 191

²⁵⁶ Klaus KANZOG: Einführung in die Editionsphilologie der neueren deutschen Literatur (Grundlagen der Germanistik, Teil 31), Berlin 1991, S. 151.

Einblendungsapparat Zuerst wurde diese Art des Apparates in der Akademieausgabe von Goethes Werken verwendet. Bei diesem Apparatmodell handelt es sich um einen linearen integralen Apparat, bei dem die Lesarten und Varianten direkt in den Editionstext einblendet werden. Für Zeile 13 aus dem Beispiel des nichtlemmatisierten Apparates sähe dies wie folgt aus:

diu[di D, Die G] herzogen[kunig g, herzogin Gd] und künege[herzoge g] riten.

Kritik an diesem Apparatmodell (vor allem bei Prosatexten) äußert vor allem Rüdiger Nutt-Kofoth in Bezug auf die Lesbarkeit.²⁵⁷ Dies ist auch der Hauptgrund, weshalb diese Methode der Variantenverzeichnung nur in wenigen Editionen Anwendung fand.²⁵⁸

Treppen- oder Stufenapparat Bei dem Treppen- oder Stufenapparat handelt es sich um einen weiteren integralen Apparatyp. Auch dieser wird direkt in den Editionstext einblendet, jedoch wird hier ganz im Sinne Backmanns die absolute Chronologie der Varianten in vertikaler und die relative in horizontaler Richtung dargestellt.²⁵⁹ Dabei können einzelne Textstufen treppenartig eingerückt vorliegen. Dieser Apparat eignet sich vor allem zur Darstellung von Textänderungen der Binnenvarianz (meist eines Autors).²⁶⁰ Handelt es sich um einen solchen binnenvarianten Text, richtet sich die „lineare Textdarstellung [...] nicht nach der räumlichen Anordnung und Verteilung der Schriftzeichen auf dem Manuskript, sondern stellt Zeile für Zeile alle Vorstufen, Varianten und Korrekturen des Textes so dar, daß sie in der vermutlichen Reihenfolge ihrer Niederschrift lesbar werden.“²⁶¹ Die treppen- oder stufenartige Anordnung text-

²⁵⁷ „Was auf der einen Seite den Schreibprozess und den Änderungsvorgang sichtlich macht, führt auf der anderen Seite aber zu einer ganz entschiedenen Reduktion der Möglichkeit, eine Fassung des Werktextes in Gänze rezipieren zu können. Zum einen muß der Benutzer laufend Text überspringen und das Anschlußstück suchen, zum anderen ist ihm immer zugleich der gesamte Text - ähnlich wie beim synoptischen Apparat, nur noch weniger gegliedert - auch optisch präsent. Die Ausgaben suchen dem Problem durch verschiedene Arten der typographischen Auszeichnungen mit dem Ziel zu begegnen, die - immer noch reichhaltigen - editorischen Zeichen zu verringern, doch bleibt festzustellen, daß für die Verzeichnung von Prosatexten nach dem integralen Verfahren der Darstellung der Textgenese der absolute Vorrang vor dem Rezeptionsbedürfnis gewährt wird.“ Rüdiger NUTT-KOFOTH: Schreiben und Lesen: Für eine produktions- und rezeptionsorientierte Präsentation des Werktextes in der Edition, in: Text und Edition. Positionen und Perspektiven 2000, hrsg. v. Rüdiger NUTT-KOFOTH u. a., S. 203–229, S. 193f.

²⁵⁸ Zu erwähnen wären hier das Arbeitstagebuch von Kloppstock (1977 von Klaus Hurlebusch): Klaus HURLEBUSCH (HRSG.): Friedrich Gottlieb Kloppstock: Werke und Briefe. Historisch-kritische Ausgabe. Band 2, Berlin, New York 1977 oder auch Eichendorfs Taugenichts (1989 von Karl Polheim): Karl POLHEIM: Text und Textgeschichte des 'Taugenichts'. Eichendorffs Novelle von der Entstehung bis zum Ende der Schutzfrist, Tübingen 1989, S. 19–50. Besonders eindrücklich schildert auch Ulrich Bubrowski die Schwierigkeiten mit Einblendungsapparaten am Beispiel Barlach. Ulrich BUBROWSKI: Editorische Schach- und Winkelzüge oder Versuch, unbeherrschbarer Schreibverhältnisse Herr zu werden - am Beispiel Barlach, in: Editio. Beihefte, Teil 10 1998, S. 117–154, S. 133ff.

²⁵⁹ URCHUEGUIA: Apparat (wie Anm. 251).

²⁶⁰ vgl. http://www.uni-bamberg.de/fileadmin/uni/fakultaeten/split_professuren/ndl/ES_NdL_I-2010_Online-Materialien_Editionsphilologie.pdf S. 3

²⁶¹ Sophia VIETOR: Astralis von Novalis. Handschrift - Text - Werk (Stiftung für Romantikforschung, Teil 15), Würzburg 2001, S. 93.

genetischer Prozesse geht auf Friedrich Beißner zurück, der diese Art der Variantendarstellung Ende der 1930er Jahre am Beispiel der Werke Hölderlins entwickelte.²⁶² Folgendes Beispiel zeigt einen Stufenapparat, der die Textgenese eines Ausschnitts aus den Philosophischen Schriften von Gottfried Wilhelm Leibniz verzeichnet.²⁶³

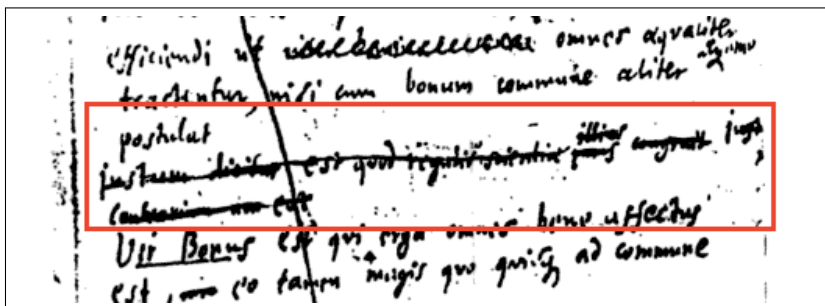


Abbildung 2.4

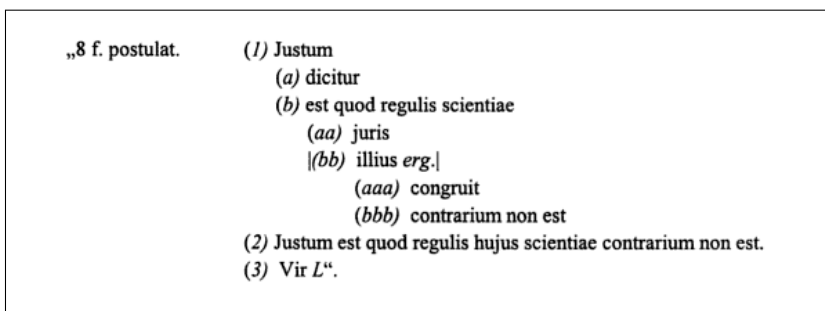


Abbildung 2.5 Treppen- oder Stufenapparat zu Abbildung 2.4

Das Lemmawort *postulat* der Zeile 8 markiert die Stelle des Textes, nach der Leibniz in verschiedenen Schreibphasen Änderungen am Text vorgenommen hat. Die erste Ebene (gekennzeichnet durch die geklammerten von eins aufsteigenden Zahlen) kennzeichnet die erste Textstufe. Stufe eins ist dabei weiter treppenartig untergliedert. Jede weitere Unterstufe wird durch ein System von kleinen Buchstaben (alphabetisch sortiert, beginnend bei a) unterschieden.²⁶⁴ Dabei beginnt die Zählung jeder Unterstufe wieder erneut bei a, jedoch immer mit einem zusätzlichen Buchstaben. (aaa ist immer die erste Variante innerhalb von bb). Dabei ersetzt ein Eintrag immer den vorherigen Eintrag der selben Stufe (bb ersetzt aa). Der letzte Eintrag einer Stufe bleibt so lange erhalten, bis er vom nächsten Eintrag der selben Stufe ersetzt wird. Auch dieser auf Beißner zurückgehende Ansatz der Variantenverzeichnung und -darstellung ist

²⁶² KANZOG: Einführung in die Editionsphilologie (wie Anm. 256), S. 155.

²⁶³ Abbildung nach Norbert HEROLD/Sibylle MISCHER (HRSG.): Philosophie. Studium, Text und Argument, Münster 2003, S. 57

²⁶⁴ Hier stellt sich allerdings die Frage, ob diese Gliederung überhaupt sinnvoll ist, da die Einrückung der Varianten ja bereits die nächstfolgende Stufe kennzeichnen.

nicht kritiklos aufgenommen worden. Bemängelt wurde in erster Linie die mangelnde Objektivität des Verfahrens.²⁶⁵ Der Leser sei nicht imstande, die Entscheidungen des Editors eindeutig nachzuvollziehen. Eine räumliche Beschreibung des handschriftlichen Befundes fehle vollständig. Das Verfahren sei auf die letzte Arbeitsstufe des Autors fixiert und beschreibe nicht den realen Arbeitsprozess des Dichters.²⁶⁶

Synoptischer Apparat Der synoptische Apparat ordnet alle Varianten verschiedener Stadien parallel so an, dass sie entweder in Zeilen- oder Abschnittssynopsen (meist untereinander) dargestellt werden. Besonders für Gedichte eignet sich die Darstellungen der Zeilensynopse.²⁶⁷ Ein wegweisendes Modell der synoptischen Variantendarstellung wurde von Hans Zeller für die historisch-kritische Conrad Ferdinand Meyer-Ausgabe entwickelt.²⁶⁸ Als Reaktion auf die Kritik an dem Beißnerschen Stufenmodell, die Editorentscheidungen nicht transparent genug nachvollziehen zu können²⁶⁹ (vor allem die unzureichende Trennung von objektivem Befund und eigener Deutung), parallelisierte er den Editionstext zeilenweise und führte eine Reihe diakritischer Zeichen ein, die „über die graphische Position der unterschiedlichen Korrekturen in der Handschrift informieren“²⁷⁰, dies solle „den Leser in die Lage versetzen, zumindest im Prinzip den handschriftlichen Befund anhand dieser Angaben zu rekonstruieren.“²⁷¹ Einen nach diesem Modell entwickelten Apparat stellt Kanzog vor (Abbildung 2.7).

Im Gegensatz zum Stufenapparat Beißners gilt hier der Grundsatz, dass in der synoptischen Darstellung kein Wort wiederholt werden darf, das nicht auch vom Schreiber selbst wiederholt wurde (zu sehen am Beispiel des Treppenapparates in 2.1.1, bei

²⁶⁵ Werner BESCH (HRSRG): Sprachgeschichte. Ein Handbuch zur Geschichte der deutschen Sprache und ihrer Erforschung. 2. vollständig neu bearbeitete und erweiterte Auflage, Berlin 1998, S. 933.

²⁶⁶ BOHNENKAMP: Editionswissenschaft (wie Anm. 241).

²⁶⁷ Vgl. NUTT-KOFOTH: Schreiben und Lesen (wie Anm. 257), S. 193. An anderer Stelle verweist Nutt-Kofoth auch auf den Zusammenhang textgenetischer Methoden und deren Bedeutung für die Verstexteditionen:

...lässt sich die Verstextedition als Paradigma einer Geschichte der neugermanistischen Edition verstehen. Das mag daran liegen, dass durch die Gliederung in Vers und Strophe, durch Metrik oder Reimbildung Hilfestellungen bei der Erschließung von Änderungszusammenhängen vorliegen, die bei Prosatexten fehlen. Zu bedenken und vielleicht durch vergleichende Untersuchungen zu bestätigen wäre auch, dass Kurztexte wie Gedichte tendenziell häufiger verändert, überarbeitet werden, sich leichter neu abschreiben lassen usw. als längere Texte, so dass zumindest für Gedichte umfangreicheres textgenetisches Material als bei anderen Textsorten vorliegen kann. Daher mag es kein Zufall sein, dass die zunehmende Akzentuierung der Textgenese in der Geschichte der neugermanistischen Edition vor allem an Verstextausgaben manifest wird.

DERS.: Edition und Interpretation Moderner Lyrik seit Hölderlin. In: Beihefte zu Editio 2010, hrsg. v. Dieter BURDORF, S. 17–39, S. 26f;

Zur Problematik der synoptischen Methode bei Prosatexten vergleiche auch: ZELLER: Typen des Germanistischen Variantenapparats (wie Anm. 249).

²⁶⁸ Vgl. DERS.: Befund und Deutung. Interpretation und Dokumentation als Ziel und Methode der Edition (wie Anm. 187).

²⁶⁹ ZWERSCHINA: Variantenverzeichnung (wie Anm. 240), S. 210f.

²⁷⁰ BOHNENKAMP: Editionswissenschaft (wie Anm. 241).

²⁷¹ Ebd.

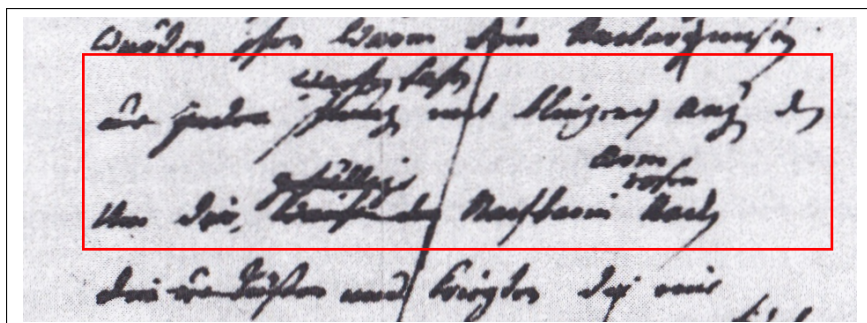


Abbildung 2.6

I 2:	a	[u. jeder]	
	b	[schlangen]	mit blinzenden Augen den Arm
	c/e \wedge	[warfen]	
	d/e	[sahen]	
I 3:	a	Um d[ie]	[Weiche der] Nachbarin
	b	er \wedge [gefälligen]	Nacken
	c		\wedge rosen

Abbildung 2.7 Synoptischer Apparat zu Abbildung 2.6

dem in der 2. Textschicht alle Entwicklungen der ersten wiederholt werden: (2) Justum est quod ...). Die eckigen Klammern stehen hier für eine Tilgung (diese muss nicht zwangsläufig durch ein typographisches Zeichen, wie etwa eine Streichung sichtbar sein). Das Zeichen \wedge in Textstufe I 2: e/c signalisiert hier, dass die Einfügung über der Textzeile vorgenommen wurde. Der nicht getilgte Text (also der nicht eingeklammerte) wird jeweils mit in die nächste Zeile genommen. Für die Stufen I 2: b-e ergibt sich für die Textgenese also die Abfolge: 1. schlangen wird getilgt (b), 2. durch warfen ersetzt (c), 3. wiederum durch sahen ersetzt (d), und 4. die Tilgung von warfen wieder zurückgenommen (e - hier angezeigt durch den kleinen Punkt unter warfen). Die kursive Schrift in diesem Beispiel zeigt den letztgültigen Text einer jeden Stufe an. Dies ist sehr praktisch, da so mit ein bisschen Übung der konstituierte Text flüssig gelesen werden kann. Während lemmatisierte Apparate immer einen Bezugstext benötigen, sind synoptische Darstellungen unabhängig von diesem.²⁷²

Aber auch bei diesem Apparatmodell ließ die Kritik nicht lange auf sich warten. Grundsätzlich wurde der Aufwand kritisiert, mit welchem die Erstellung solcher Apparate verbunden war. Um den Befund von der Deutung sauber zu trennen, musste eine große Zahl von diakritischen Zeichen eingeführt werden und bei schwieriger oder unklarer Überlieferungslage war man schließlich doch wieder gezwungen, auf einen lemmatisierten Zweitapparat zurückzugreifen.²⁷³ Zwerschina kritisiert das Ziel der

²⁷² Vgl. NUTT-KOFOTH: Edition und Interpretation (wie Anm. 267), S. 26.

²⁷³ Vgl. dazu auch Herbert KRAFT: Editionsphilologie, Darmstadt 1990, S. 155.

Methode, mit der synoptischen Darstellung eine „metasprachliche Wirklichkeit“²⁷⁴ des Schaffensprozesses suggerieren zu wollen, die seiner Meinung nach so nicht existiere.²⁷⁵ Nutt-Kofoth warnt hingegen vor einer möglichen (und teilweise zweifelhaften) Überflutung von Textfassungen.²⁷⁶

2.1.2 Faksimile

Neuere Entwicklungen in den Editionswissenschaften lassen eine teilweise Abkehr von den ursprünglichen Forderungen Backmanns und Zellers, eine Handschrift durch genaue Beschreibung mit Hilfe textgenetischer Apparate wiederherstellbar machen zu können, erkennen.²⁷⁷ So äußern sich schon 1969 die Autoren der Georg Trakl-Ausgabe durchaus kritisch zum Rekonstruierbarkeitsparadigma: „Wir waren frei von dem Ehrgeiz, die Handschrift durch ein System von Zeichen vollständig rekonstruierbar zu machen. Das hätte uns Entscheidungen abgenötigt, die aus der Genauigkeit des Philologen Willkür machen.“²⁷⁸ Nutt-Kofoth verweist in diesem Zusammenhang auch auf die noch weiter zurückliegende Einschätzung Karl Goedeke in seinem Vorwort zum dramatischen Nachlass Schillers.²⁷⁹ Karl Goedeke bemerkt schon 1876: „Nur eine photographische Wiedergabe könnte einen Begriff gewähren, was dem Dichter während der Arbeit der Aufzeichnung bedürftig erschien. Aber auch nur in der Photographie würde die Art seines eigentlichen Schaffens deutlich werden. Dazu reichen gestrichene Lettern und Schriftsorten nicht aus.“²⁸⁰ Faksimile-Ausgaben schienen nun mehr und mehr in das Blickfeld der Editionswissenschaft zu rücken, da dadurch am ehesten die Forderung nach objektiver Überprüfbarkeit der Editorentscheidungen eingelöst werden konnte. Die Faksimiledarstellung bindet den Benutzer näher an den eigentlichen

²⁷⁴ ZWERSCHINA: Variantenverzeichnis (wie Anm. 240), S. 211f.

²⁷⁵ „Auch wenn man den Vergleich der Partitur [von Zeller selbst angestellter Vergleich mit seinem synoptischen Modell] weniger wörtlich nimmt und das 'Ganze' als 'Sinn-Ganzes' versteht, das sich als Zusammenklingen aller einzelnen Entstehungsstufen ergibt, dann setzte dies voraus, daß den Syntagmen der einzelnen Entstehungsstufen ein gemeinsamer 'Sinn' zu eigen ist, daß die Einzelstimmen der Partitur unter dem Stern von etwas a priori Fixiertem, etwas der 'Idee' (oder eben dem 'Sinn') stünden. Dann wäre Textproduktion als Annähern an eine metasprachliche Wirklichkeit zu verstehen.“ ebd., S. 211f.

²⁷⁶ „Der Umgang mit den Druckfassungen der Texte weist aber auf einen grundsätzlichen Problem-bereich hin, der sich durch die synoptische Präsentationsmöglichkeit ergibt. Setzt man voraus, dass es neben der Textdynamik zugleich die statischen Momente des Werktextes gibt, nämlich die verschiedenen, separat lesbaren Fassungen des Werkes innerhalb seiner Genese, dann führt die synoptische Darstellung zu einer latenten Multiplikation von Fassungen, insbesondere wenn man den rigiden strukturalistisch begründeten Fassungs-begriff in Rechnung stellt, nach dem sich eine neue Fassung durch eine einzige Variante konstituieren kann.“ Vgl. NUTT-KOFOTH: Edition und Interpretation (wie Anm. 267), S. 25.

²⁷⁷ Vgl. Hans ZELLER: Fünfzig Jahre neugermanistische Edition. Zur Geschichte und künftigen Aufgaben der Textologie, in: Editio 3 1989, S. 1–17, S. 5; sowie HURLEBUSCH: Hermeneutik (wie Anm. 216), S. 25.

²⁷⁸ Walther KILLY/Hans SZKLENAR (HRSG.): Georg Trakl: Dichtungen und Briefe. Band 2, Salzburg 1969, S. 9.

²⁷⁹ Vgl. NUTT-KOFOTH: Edition und Interpretation (wie Anm. 267), S. 27.

²⁸⁰ Karl GOEDEKE (HRSG.): Schillers sämtliche Schriften. Historisch-kritische Ausgabe, Bd. 2, Stuttgart 1876, S. 6f.

Edierprozess. Er kann Lesarten des Editors direkt nachvollziehen.²⁸¹ Der Überprüfbarkeit einer Textkritischen Edition am Original wird somit Rechnung getragen und schafft eine Transparenz, die der edierte Text alleine nicht zu leisten vermag. Für Nutt-Kofoth liegt der eigentliche Mehrwert textgenetischer Editionen dann auch in der kombinierten Präsentation von genetischem Apparat und Faksimile: „Indem die Faksimiles die Textgenese visualisierend begleiten, machen sie schon die nicht immer leicht lesbare textgenetische Darstellung schneller verständlich.“²⁸² Des weiteren betont er die Rolle des Faksimile als „(reduzierte) Reproduktion des Archivs.“²⁸³ Diese Einsicht ist natürlich besonders auf die technischen Entwicklungen im Bereich der Digitalen Editionen zurückzuführen. Der aktuell anhaltende Trend zu einer quellen-nahen Überlieferung ist zu einem großen Teil den Möglichkeiten der digitalen Faksimilierung zu verdanken. Das zu Zeiten der klassischen Druckausgaben oftmals noch ausschlaggebende Kriterium der hohen Druckkosten entfällt bei Onlinepublikationen beispielsweise nahezu vollständig. Digitale Faksimileausgaben können heutzutage in qualitativ hochwertiger Form kostengünstig hergestellt und einem breiten Publikum online zugänglich gemacht werden.²⁸⁴

2.1.3 Paralleldruck

Paralleldrucke gehen in erster Linie auf Editionsverfahren der Bibelwissenschaften zurück.²⁸⁵ Auch wenn Paralleldrucke nicht zu den klassischen Apparatypen gehören, bieten sie doch die Möglichkeit, textgenetische Prozesse zu visualisieren und bilden in gewisser Weise die Vorstufe zur Darstellung von Abschnitts- oder Seitensynopsen im digitalen Medium. Für die Darstellung von Überlieferungsvarianten ist der Paralleldruck (mit und ohne graphischer Kennzeichnung) daher auch des öfteren in klassischen Printeditionen zum Einsatz gekommen. Einfache Paralleldrucke setzen die Texte lediglich zueinander in Beziehung, während markierte Paralleldrucke sogenannte Variantensignale bieten.²⁸⁶

Insbesondere bei mittelalterlichen Texten ist die Authentizität von unterschiedlichen Fassungen nicht immer eindeutig zu klären, etwa dann, „wenn Varianten nicht

²⁸¹ Im besonderen Maße gilt dies natürlich für Digitale Editionen: „A major advantage of image-based electronic editions over print editions is that the display of the text is transparent“ Kevin KIERNAN: Digital Faksimiles in Editing, in: *Electronic Textual Editing 2006*, hrsg. v. Lou BURNARD/Katherine O'BRIEN O'KEEFFE/John UNSWORTH, S. 262–268, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/kiernan.xml, hier S. 265.

²⁸² Vgl. NUTT-KOFOTH: *Edition und Interpretation* (wie Anm. 267), S. 28.

²⁸³ Ebd., S. 28.

²⁸⁴ Zwerschina führt entsprechend auch die folgenden Gründe für das wachsende Interesse an den originalgetreuen Reproduktionen von Textvorlagen an: „Neue technische Möglichkeiten, der Zwang, editorische Entscheidungen überprüfbar zu machen und der Wunsch, die Dokumentation des Befundes aussagekräftiger und benutzerfreundlicher zu gestalten.“ ZWERSCHINA: *Variantenverzeichnung* (wie Anm. 240), S. 209. Vgl. auch DERS.: *Von den Faksimiles zur Interpretation*, in: *Perspectives of Scholarly Editing / Perspektiven der Textedition*. Hg. von H.T.M. van Vliet und Bodo Plachta. 2002.

²⁸⁵ KANZOG: *Einführung in die Editionsphilologie* (wie Anm. 256), S. 153.

²⁸⁶ Ebd., S. 153.

nur vereinzelt auftreten, sondern ganze Strophen in ihrer Substanz betreffen.“²⁸⁷ Daraus kann abgeleitet werden, dass es nicht immer einfach ist, die authentischste Variante eines Textes zu finden. Dann allerdings ist es auch nicht möglich, einen Text als Ursprung im Sinne des Lachmannschen Archetyps zu favorisieren. Auch um nicht das Zellersche Paradigma der Trennung von Befund und Deutung²⁸⁸ zu verletzen, wäre eine 'gleichberechtigte' Verzeichnung und Darstellung der Varianten vorzuziehen. Schließlich sollte die Interpretation (Deutung) eines Befundes ja immer durch den Leser überprüfbar bleiben.²⁸⁹ Bein schlägt hier als eine mögliche Alternative zur Verzeichnung der Varianten in Apparaten die Darstellung in Parallelabdrucken vor: „Der Umgang mit Fassungsvarianten (Mehrfachfassungen) dürfte abhängig sein vom Konzept und Anspruch der Edition. Fassungen sollten nach Möglichkeit nicht 'zertrümmert' in Lesartenapparaten verschwinden, sondern z.B. als Parallelabdruck erscheinen, um allererst weiter Diskussionen um die Gründe für solche Phänomene zu ermöglichen.“²⁹⁰ Auch Kanzog sieht in diesem Verfahren der parallelen Darbietung der Varianten wesentliche Vorteile gegenüber den Apparatmodellen (Abbildung 2.8).²⁹¹

1. Paralleldruck der ersten Strophe:	
<i>Es sah'</i> ein Knab' ein Röslein stehn,	Sah ein Knab' ein Röslein stehn,
<i>Ein</i> Röslein auf der Heiden,	Röslein auf der Heiden,
<i>Er sah es</i> war so frisch und schön	War so jung und morgens schön,
<i>Und blieb stehn, es anzusehn</i>	Lief er schnell es nah zu sehn,
<i>Und stand in süßen</i> Freuden.	Sah's mit vielen Freuden.
Röslein, Röslein, Röslein, roth,	Röslein, Röslein, Röslein roth,
Röslein auf der Heiden.	Röslein auf der Heiden.

Abbildung 2.8 Markierter Paralleldruck mit kursiver Kennzeichnung der Varianz

2.2 State-of-the-art Projekte

Im folgenden sollen in Kürze die wichtigsten Projekte vorgestellt werden, die einen größeren Einfluss auf die Entwicklung Digitaler Editionen im Allgemeinen und auf textgenetische Fragestellungen im Speziellen ausübten. Hier sind vor allem das „Canterbury Tales Project“²⁹², „Wittgenstein's Nachlass - the Bergen electronic edition“, „Hypertextuelle“²⁹³ und die Elektronische Musil-Edition²⁹⁴ zu nennen.

²⁸⁷ BEIN: Textkritik (wie Anm. 232), S. 141.

²⁸⁸ Vgl. ZELLER: Befund und Deutung. Interpretation und Dokumentation als Ziel und Methode der Edition (wie Anm. 187), S. 45-90.

²⁸⁹ Vgl. ebd., S. 79ff.

²⁹⁰ BEIN: Textkritik (wie Anm. 232), S. 143.

²⁹¹ KANZOG: Einführung in die Editionsphilologie (wie Anm. 256), S. 152.

²⁹² <http://www.canterburytalesproject.org/>

²⁹³ <http://www.nietzschesource.org>

²⁹⁴ <http://wwwg.uni-klu.ac.at/musiledition/>

Das **Canterbury Tales Project**²⁹⁵ war eines der ersten Editionsprojekte, das sowohl bei der Datenerfassung, der Analyse und der Publikation²⁹⁶ konsequent auf softwaretechnologische Methoden setzte. Das Projekt war insofern wegweisend, dass es wichtige Impulse im Bereich des automatischen Vergleichs von Textvarianten setzte, die aufgrund ihrer Überlieferungsmenge nicht mehr manuell zu analysieren gewesen wären. Initiiert von Peter Robinson in den späten 80er Jahren untersuchte es die Verwandtschaftsverhältnisse der ca. 80 noch erhaltenen Manuskriptversionen der 'The Canterbury Tales' von Geoffrey Chaucer. Das Projekt verwendete bei der Analyse der Textzeugen Algorithmen, die aus der Evolutionsbiologie stammen und dort zur Erstellung Phylogenetischer Bäume genutzt werden.²⁹⁷ Aus den Sequenzanalysen von 58 Textzeugen konnten so Hypothesen über deren Verwandtschaftsverhältnisse aufgestellt und die Vermutung nahe gelegt werden, dass der Ursprung dieser Abschriften, Chaucers 'Original', keine fertige Version, sondern ein textgenetischer Entwurf war, der vermutlich zahlreiche alternative Abschnittsfassungen und Anweisungen zum Streichen oder Hinzufügen von Passagen enthielt. Trotz der überwiegend positiven Resonanz, die dieser Ansatz in der Fachwelt hervorrief, blieben aber auch kritische Stimmen nicht aus. So gibt z.B. Hanna zu bedenken, dass diese Methode die Kontaminationen von Texten nicht berücksichtigt, die bei der Abschrift von mehr als einer Textvorlage entstünden.²⁹⁸ Neuere Ansätze versuchen dies zu berücksichtigen und betonen dabei die Parallelen zwischen Textkontamination und der Neukombination genetischer Sequenzen in biologischen Organismen.²⁹⁹ Aus den Arbeiten an diesem Projekt gingen unter anderem das Kollationierungswerkzeug Collate, der Vorgänger

²⁹⁵ Peter ROBINSON: The History, Discoveries and Aims of the Canterbury Tales Project, in: Chaucer Review (38.2), S. 126–139, URL: <http://www.canterburytalesproject.org/pubs/PR-ChauRev.pdf>.

²⁹⁶ Publiziert wurden die Transkriptionen auf CD-Rom. Aufgrund der Datenmenge sah man von einer Print-Publikation ab. ROBINSON/TAYLOR: Publishing (wie Anm. 112), S. 275

²⁹⁷ Vgl. C. J. HOWE/Peter M. W. ROBINSON: The Phylogeny of The Canterbury Tales, in: Nature 394, URL: www.canterburytalesproject.org/pubs/nature.pdf und Robert O'HARA/Peter ROBINSON: Computer-assisted methods of stemmatic analysis. Canterbury Tales Project, in: Occasional Papers I 1993, S. 53–74, URL: www.canterburytalesproject.org/pubs/op1-cladistics.pdf Den Nutzen dieser Methoden zeigen Windram et al. auch anhand eines Vergleichs mit traditionellen stemmatologischen Analysen am Beispiel Dante's 'Monarchia' Heather F. WINDRAM/Prue SHAW/Peter ROBINSON: Dante's Monarchia as a test case for the use of phylogenetic methods in stemmatic analysis, in: Literary and Linguistic Computing (23.4) 2008, S. 443–463 und auch Heikkilä TUOMAS/Teemu ROOS: Evaluating methods for computer-assisted stemmatology using artificial benchmark data sets, in: Literary and Linguistic Computing (24.4) 2009, S. 417–433.

²⁹⁸ Ralph HANNA: The application of thought to textual criticism in all modes - with apologies to A. E. Housman, in: Studies in Bibliography 53 2000, S. 163–172, Vgl. Allgemeinere Bedenken äußert auch Stolz: "Es stellt sich nun [...] selbstverständlich die Frage, ob es ausreicht, handschriftliche Kopiervorgänge mit Befunden der Genetik zu vergleichen, oder ob in Texttraditionen nicht weitere Aspekte ins Spiel kommen, die über molekularbiologische Prozesse hinausführen. Wo zeichnen sich bei der Wiedergabe einer Vorlage durch Schreiber Abweichungen, Widerstände und Widersprüche ab, die eine bloße mechanistische Aneignung übersteigen? Inwiefern ist das Wiederschreiben [...] von Text 'materieller' Vorgang, inwiefern ist es Reproduktion und damit auch Produktion von Sinn?" Michael STOLZ: Datenaufbereitung für elektronische Publikationen, in: Materialität in der Editionswissenschaft 2010, hrsg. v. DEMS., S. 275–291, 280f.

²⁹⁹ Heather F. WINDRAM/Christopher J. HOWE/Matthew SPENCER: The identification of exemplar change in the Wife of Bath's Prologue using the maximum chi-squared method, in: Literary and Linguistic Computing (20.2) 2005, S. 189–204, Vgl.

des in Abschnitt 1.2.4.3 schon erwähnten CollateX, und die Publikationssoftware Anastasia hervor.

Wittgenstein's Nachlass, die **Bergen Electronic Edition**³⁰⁰ besteht aus ca. 20.000 Manuskriptseiten und war ebenso von Beginn an als Digitale Edition angelegt worden. Charakteristisch für Wittgensteins Arbeitsweise war das interlineare Einfügen von Textabschnitten, die häufig mehr als eine Lesart zuließen. Darüber hinaus beachtenswert sind sein ausgefeiltes System von Verweisen innerhalb und außerhalb des Textes, sein semantisches System aus verschiedenen Formen des Unterstreichens und das Kombinieren von Texten aus unterschiedlichen Manuskripten. Seine Überlieferungen beinhalten zahlreiche inter- und intratextuelle Bezüge von komplexer Struktur. Da es sich bei dem Nachlass im Wesentlichen um ähnliche Textfragmente aus unterschiedlichen Textzeugen handelt, wurde ein synthetisches Verfahren angewendet, das Varianten auf der Grundlage eines Basistextes verzeichnet. Von der ursprünglichen Idee, eine dokumentarische Edition im Sinne einer Volltranskription aller Textzeugen vorzulegen, nahmen die Autoren schnell Abstand, da die Verzeichnung von größtenteils wortgleichen Textzeugen und die Verknüpfungen dieser zahlreichen Varianten zu einer zu großen Komplexität der Datenstruktur und zu einem zu großen Arbeitsaufwand geführt hätten. Wie zeitaufwändig die Arbeiten an einem solchen Projekt sind, wird deutlich, wenn man sich die Kalkulation der Autoren vor Augen führt, die für die Transkription nach eigenen Angaben ca. 10 Jahre benötigten, was einem Arbeitsaufwand von ca. 40 Personenjahren entsprach.³⁰¹ Bemerkenswert an diesem Projekt war die Verwendung einer eigens konzipierten Auszeichnungssprache mit Namen MECS (Multi-Element Code System).³⁰² Diese wurde entwickelt, um den komplexen Anforderungen multipler Hierarchien gerecht zu werden, die zur damaligen Zeit von SGML nur unzureichend unterstützt wurden.³⁰³ Später wurde MECS zu MLCD (Markup Language for Complex Documents)³⁰⁴ weiterentwickelt. MLCD versucht dabei die Vorteile von XML und MECS zu verbinden, indem es die Möglichkeit der Kodierung multipler Hierarchien von MECS und die Notation, Datenstruktur und Grammatik von XML übernimmt. Auch im Bereich der Texttheorie sind wichtige Impulse aus diesem Projekt gekommen.³⁰⁵ Neuere Ansätze befassen sich mit ontolo-

³⁰⁰ Vgl. Claus HUITFELDT: Editorial principles of Wittgenstein's Nachlass the Bergen electronic edition, in: *Augmenting Comprehension: Digital Tools and the History of Ideas 2004*, hrsg. v. D. BUZZETTI/G PANCALDI/Short H., S. 113–128.

³⁰¹ Ebd., S. 116.

³⁰² Vgl. HUITFELDT: MECS (wie Anm. 98) und CZMIEL: Markupsysteme (wie Anm. 74), 13ff.

³⁰³ HUITFELDT: Editorial principles (wie Anm. 300).

³⁰⁴ <http://xml.coverpages.org/mlcd.html>

³⁰⁵ Vgl. Alois PICHLER: Wittgensteins Philosophische Untersuchungen: Vom Buch zum Album, in: *Studien zur Österreichischen Philosophie 36 2004*, hrsg. v. R. HALLER. Pilcher betont in seiner Theorie besonders die gleichermaßen Abhängigkeit des Textbegriffs von Autor und Leser: „Texts are not objectively existing entities which just need to be discovered and presented, but entities which have to be constructed. They are products of both the author and the reader“ DERS.: *Encoding Wittgenstein. Some remarks on Wittgenstein's Nachlass, the Bergen Electronic Edition, and future electronic publishing and networking*, in: *TRANS. Internet-Zeitschrift für Kulturwissenschaften 10 2002*, URL: <http://www.inst.at/trans/10Nr/pichler10.htm>. Die innovativen Ideen bezüglich der technischen und theoretischen Konzepte innerhalb des Projekts sind in der Fachwissenschaft durchweg positiv aufgenommen worden. Joachim SCHULTE: Witt-

giebasierten semantischen Analysen, um einen tieferen Einblick in die philosophische und intertextuelle Arbeitsweise Wittgensteins zu erhalten.³⁰⁶ Seit 2001 setzt das Projekt vermehrt auf die Auszeichnungsrichtlinien der TEI.³⁰⁷ Ursprünglich erschien die Edition als CD-Rom in Verbindung mit der Publikationssoftware Folio-Views. 2005 sind die ersten 5000 Seiten des Nachlasses auch im WWW publiziert worden. Sie besteht aus 3 Hauptkomponenten: Den Faksimile, der diplomatischen Transkription und einer normalisierten Transkription. Die diplomatische Transkription enthält dabei die ganze Bandbreite an Textkodierung: Streichungen, (interlineare) Hinzufügungen, Randbemerkungen, Ersetzungen, die unterschiedlichsten Metamarkierungen in Form von Kürzeln, Positionsangaben etc. Die Transkriptionen bestehen aus relativ statischen HTML-Seiten und kennzeichnen die Modifikationen mit farbigen Texthinterlegungen, Streichungen, Hochstellungen des Textes etc. Sie sind nicht mit dem Digitalisat verknüpft und es stehen keine weiteren Werkzeuge zur Verfügung, um den genetischen Entstehungsprozess der Manuskripte nachvollziehen zu können.³⁰⁸ Eine genauere Analyse folgt im Abschnitt 2.4.

Ähnliches gilt auch für die **Musil-Edition**.³⁰⁹ Wegen ihres enormen Umfangs (ca. 10.000 Manuskriptseiten) wurde auch hier auf eine Druckversion verzichtet. Charakteristisch für Musils Arbeitsweise ist seine immer wiederkehrende Kombination aus unterschiedlichen Notiz- und Entwurfsblättern (sogenannten Schmierblättern) zu einem abschließenden Entwurfsmanuskript.³¹⁰ Fanta unterstreicht die Bedeutung des Produktionsablaufs dieser Kombinationstechnik Musils für seinen kreativen Schaffensprozess.³¹¹ Auch dieses Projekt bediente sich der Software Folio-Views und veröffentlicht die Edition bis heute ausschließlich auf CD bzw. DVD.³¹² Die Resonanz in der Fachwelt auf die Edition war durchweg positiv.³¹³ Analog zur Bergen Electro-

genstein's Nachlass: The Bergen Electronic Edition, in: Grazer philosophische Studien 65 2002, S. 237–246, Vgl. Bis heute haben diese Entwicklungen einen großen Einfluss auf Forschungsgebiete, wie die der Markup-Technologie oder auch der Texttheorie ausgeübt. Vgl. SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 49, 273.

³⁰⁶ Vgl. A. PICHLER/A. ZÖLLNER-WEBER: Towards Wittgenstein on the Semantic Web, in: *Chaucer Review* (38.2) 2012, S. 318–321, URL: <http://www.dh2012.uni-hamburg.de/conference/programme/abstracts/towards-wittgenstein-on-the-semantic-web/> und C. ERBACHER: Unser Denken bleibt gefragt: Web 3.0 und Wittgensteins Nachlass, in: *Wissenschaftstheorie, Sprachkritik und Wittgenstein* 2011, hrsg. v. S. WINDHOLZ/W. FEIGL, S. 135–146 Auch ein Ontologie-Annotationstool namens SwickyNotes ist in diesem Zusammenhang entstanden C. MORBIDONI/M. NUCCI: SwickyNotes user guide, 2010, S. 237–246, URL: www.swickynotes.org/docs/SWickyNotesStartingGuide.pdf.

³⁰⁷ PICHLER/ZÖLLNER-WEBER: *Semantic Web* (wie Anm. 306).

³⁰⁸ Beispieltranskription: http://www.wittgensteinsource.org/Ms-140,39v%5B1%5D_d

³⁰⁹ Vgl. WALTER FANTA: Robert Musil – Klagenfurter Ausgabe. Eine historisch-kritische Edition auf DVD, in: *Editio* 24 2010, S. 117–148. <http://www.wg.uni-klu.ac.at/musiledition/>

³¹⁰ DERS.: Die Entstehungsgeschichte des 'Mann ohne Eigenschaften' von Robert Musil, Wien 2000, 59ff.

³¹¹ „Die Schmierblätter sind unmittelbare Zeugen des Vorgangs der Kristallisation von Gedanken und Vorstellungen in Schrift innerhalb des Produktionsablaufs Musils. Als Quelle für das Studium der Textgenese und der Arbeitsweise Musils sind sie ohne Zweifel aufschlussreich.“ ebd., S. 59.

³¹² seit 2013 auch als USB-Stick

³¹³ Vgl. DIRK GÖTTSCHE: Ausgabentypen und Ausgabenbenutzer, in: *Text und Edition. Positionen und Perspektiven* 2000, hrsg. v. RÜDIGER NUTT-KOFOTH u. a., S. 37–64, hier S. 61 und KAI LUEHRS:

nic Edition bieten die diplomatischen Textansichten auf der DVD keine dynamische Funktionalität, um die Textgenese direkt am Faksimile nachvollziehen zu können.³¹⁴

Ein weiteres Projekt, aus dem ein eigenes Markup-System für die Kodierung textgenetischer Prozesse entstand, war das **HyperNietzsche-Projekt**.³¹⁵ Die HyperNietzsche Markup Language (HNML) wurde zuerst als TEI P4-Erweiterung implementiert und ermöglichte unter anderem die Verzeichnung von dynamischen Geneseprozessen in einer Art Layersystem.³¹⁶ HNML bildete die Grundlage des seit 2010 in das TEI-Manuscript-Modul integrierten textgenetischen Kodierungsmodells für Digitale Faksimile (TEI P5 Section 11 Representation of Primary Sources³¹⁷). Innerhalb dieses Projekts sind eine Reihe von Studien zum Thema Visualisierung von Geneschichten entstanden. So z.B. das von Saller vorgestellte Layersystem, welches das stufenweise Einblenden chronologisch aufeinander folgender Schreibschichten in der HTML-Transkription ermöglichen soll,³¹⁸ oder das von Gerike stammende Konzeptpapier, das die Möglichkeiten beschreibt, textgenaue Überblendungen von Transkriptionen auf dem Digitalisat umzusetzen.³¹⁹ Eingang in die Digitale Edition haben die in den Studien entwickelten Visualisierungskonzepte aber nicht gefunden. Was alle Projekte gemeinsam haben, ist die Tatsache, dass die textgenetische Überlieferung zu umfangreich und zu komplex ist, als dass sie in einer herkömmlichen Print-Version adäquat hätte dargestellt werden können, und dass viel Potential verschenkt worden wäre, wenn man sich nicht dazu entschlossen hätte, die Publikationsform der Digitalen Edition zu suchen.³²⁰

Im Folgenden wird auf die zentralen Herausforderungen aktueller digitaler textgenetischer Editionen in Bezug auf Editionspraxis, Datenkodierung und Publikation der Endprodukte in Form von WWW-Edition eingegangen. Anhand eines Beispielprojekts, das von dem Autor dieser Arbeit konzeptionell entwickelt und technisch betreut wurde, sollen einige praktische Schwierigkeiten aufgezeigt werden, die bei

Verwirklichung oder Entzweiung? Zur Edition des Musil-Nachlasses auf CD-ROM, in: Editio 8 1994, S. 158–172, hier S. 158.

³¹⁴ siehe z.B.: 'FANTA: Immortalität (wie Anm. 10), §48 Auch hier werden die Modifikationen ähnlich wie im Wittgensteinprojekt durch Formatierungen und diakritische Zeichen des linearen Textflusses realisiert. Eine Analyse folgt in Abschnitt 2.4.

³¹⁵ <http://www.hypernietzsche.org>

³¹⁶ Harald SALLER: HNML – HyperNietzsche Markup Language, in: Jahrbuch für Computerphilologie 5 2003, S. 183–190, URL: <http://computerphilologie.digital-humanities.de/jg03/saller.html>.

³¹⁷ <http://www.tei-c.org/release/doc/tei-p5-doc/de/html/PH.html>. Vgl. auch TEI: Workgroup on Genetic Editions. An Encoding Model for Genetic Editions, 2010, URL: <http://www.tei-c.org/Activities/Council/Working/tcw19.html>.

³¹⁸ SALLER: HNML (wie Anm. 316).

³¹⁹ Inga GERIKE: Der Wanderer und sein Schatten. Manuskripte und Genese, ohne Datum, URL: <http://www.hypernietzsche.org/events/lmu/gerike-1.html>.

³²⁰ Pilcher beschrieb dies für den Wittgenstein-Nachlass folgendermaßen: „I do not want to say that book editions of Wittgenstein's Nachlass make no sense. But editing the entire Nachlass in book form is an undertaking which would not only cost too much in relation to what it would achieve, but would also be confronted with many difficult editorial problems, for which today's computer technology can provide simple solution.“ Alois PICHLER: Advantages of a Machine-Readable Version of Wittgenstein's Nachlass, in: Culture and Value. Beiträge des 18. Internationalen Wittgenstein Symposiums 1995, hrsg. v. Johannessen KJELL S./Tore NORDENSTAM, S. 770–776, URL: wab.uib.no/alois/pichler-kirchb95b.pdf, hier S. 774.

dem Versuch aufgetreten sind, textgenetische Prozesse mit Methoden zu kodieren, die abseits der üblichen Markup-Theorien (wie die der TEI) ein intuitiveres, komfortableres Edieren ermöglichen sollten. Anschließend werden an weiteren Beispielen (u. a. Wittgenstein und Musil) die Problembereiche der topographischen Kodierung von Schreibräumen und der Modellierung chronologischer Prozesse vertieft. Die so gewonnen Erkenntnisse bilden die Grundlage für die in dieser Arbeit entwickelten Konzepte, die in einem modernen, kollaborativen, browserbasierten Arbeitsumfeld zu einer effizienteren Editionspraxis und einer verbesserten Analyse und Visualisierung textgenetischer Prozesse führen sollen.

2.3 Erfahrungsbericht

Die Entwicklung der in dieser Arbeit implementierten browserbasierten Applikation zur Kodierung und Darstellung textgenetischer Prozesse geht unter anderem auf Ideen und Erfahrungen zurück, die innerhalb des Editionsprojekts „Kritische Online-Edition der Nuntiaturberichte Eugenio Pacelli (1917 - 1929)“³²¹ gesammelt wurden (siehe auch Abschnitt 2.1). Die zentrale Zielsetzung bestand darin, die Textgenese der Nuntiaturberichte vom ersten Entwurf bis zu ihrer durch Pacelli autorisierten Endfassung zu edieren. Häufig waren an dem Prozess mehrere Personen beteiligt. In der Regel waren es Pacelli selbst und seine persönlichen Sekretäre, die diese Berichte hand- oder maschinenschriftlich (oft unter Zuhilfenahme von Stenotypisten) zu Papier brachten. Dabei konnten die Berichte mehrmals den Bearbeiter wechseln und nach unterschiedlichen Korrekturschritten auch wieder an eine Person einer früheren Bearbeitungsphase, die erneut Korrekturen vornahm, zurück gereicht werden. Daraus ergab sich die Möglichkeit, nicht nur die einzelnen Eingriffe in die Dokumente zu edieren, sondern diese auch den jeweiligen Bearbeitern zuzuordnen. Dies war deswegen von großem Interesse, da somit die Urheberschaft forschungsrelevanter Textpassagen eindeutig geklärt werden konnte (Beispiel in Abbildung 2.1). Es lag also nahe, die dynamischen Prozesse sichtbar zu machen und den Benutzern der Edition die Möglichkeit zu geben, die Genese in ihren einzelnen Entstehungsschichten nachzuvollziehen. Da die Edition einen Großteil der Anforderungen in Bezug auf Kodierungspraxis und Visualisierung enthält, die aktuell an digitale textgenetische Editionen gestellt werden und für die noch keine umfassenden Problemlösungsstrategien existieren, soll die editorische Praxis und die browserbasierte Visualisierung anhand eines Beispiels im Folgenden beschrieben werden.

Das in diesem Projekt verwendete genetische Editionsmodell sollte in der Lage sein, unterschiedliche Bearbeitungsarten (Streichung, Ersetzung, Hinzufügung, etc.) zu kodieren und die einzelnen Bearbeitungsschritte in Form von Bearbeitungsphasen einer Person zuordnen zu können. Die einzelnen Schichten mussten separat annotiert werden können, um ein dynamisches Ein- und Ausblenden in einer festgelegten chronologischen Abfolge zu ermöglichen. Um den Arbeitsaufwand in Grenzen zu halten, wurde nach einer Methode gesucht, die genetischen Prozesse möglichst einfach und

³²¹ <http://www.pacelli-edition.de/>

übersichtlich kodieren zu können. Die gängige Praxis, textgenetische Prozesse in klassischen Printeditionen von Nuntiaturberichten darzustellen, ist deren Verzeichnung in genetischen Apparaten, wie das Beispiel in Abbildung 2.9 zeigt.³²² In einer ersten

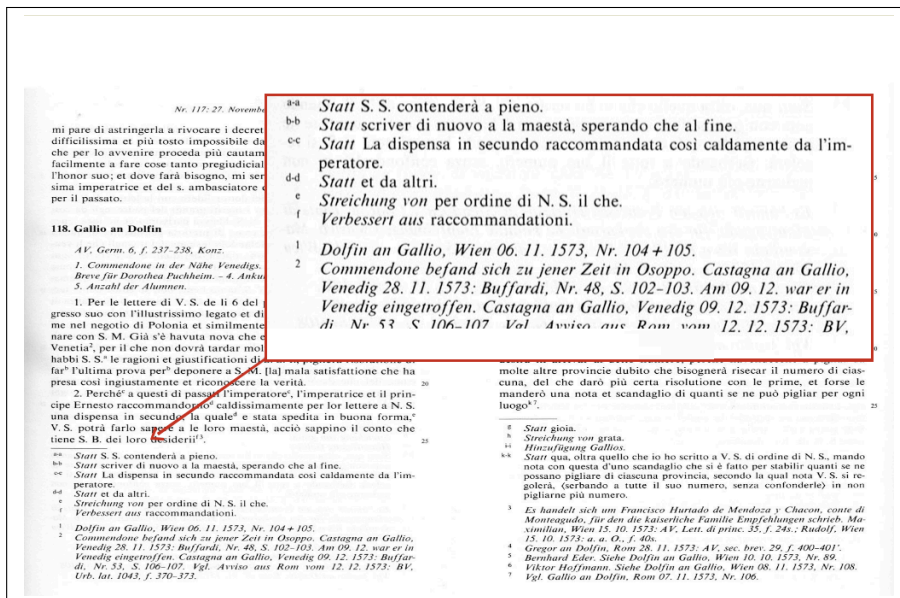


Abbildung 2.9 Textkritischer Apparat

Projektphase wurden alle Kodierungs-Funktionen eines solchen Apparates analysiert und um die benötigten zusätzlichen Funktionen (Biographische Daten, Schlagwörter etc.) ergänzt. Als brauchbarste Lösung für die Verzeichnung der textgenetischen Informationen stellte sich schließlich ein gestuftes Layersystem heraus, welches sowohl eine relativ einfache Kodierung der genetischen Prozesse ermöglichte, als auch alle gewünschten Visualisierungseffekte im Webbrowser zuließ. Aus praktischen Gründen wurde für die Aufnahme der Daten Microsoft Word in der Version 2003 gewählt, da schon ein erheblicher Teil der Berichte in Word vorlag und ab der Version 2003 ein einfacher XML-Editor zum Funktionsumfang der Software gehörte. Mit Hilfe von XML-Schemas ließen sich selbst definierte Auszeichnungselemente in das von Word verwendete XML-Format WordprocessingML (Word-ML) integrieren. Somit ließen sich die schon vorhandenen Informationsstrukturen, wie unterschiedliche Formatierungen, Tabellen, Listen, Fußnoten etc. nutzen und gleichzeitig ein projektspezifisches, XML-basiertes Editionsmodell verwenden. Für die Verzeichnung der einzelnen Bearbeitungslayer wurde aus Gründen der Übersichtlichkeit ein aus unterschiedlichen Farben bestehendes Kodierungssystem gewählt. Da es sich bei diesem Projekt um einen überschaubaren Personenkreis handelte, konnte jeder Person, die Änderungen an den

³²² aus: Nuntiaturberichte aus DEUTSCHLAND: III. Abteilung: 1572-1585, 7. Bd.: Nuntiatur Giovanni Dolfins (1573-1574), bearb. im Auftrag des Deutschen Historischen Instituts in Rom von Almut Bues, Tübingen 1990.

Nuntiaturreportagen vornahm, eine eigene Farbe zugewiesen werden. Die Möglichkeit, den Text mit ‘rohen’ XML-Strukturen auszuzeichnen wurde nach ersten Versuchen schnell verworfen, da dies bei den Projektmitarbeitern nur auf geringe Zustimmung stieß.³²³ Die große Zahl unterschiedlicher Markup-Elemente zur Kodierung der einzelnen Schichten ohne visuelle Unterscheidungskriterien verhinderte ein effizientes Verzeichnen der genetischen Strukturen. Die in Word verwendeten Textfarben und das versteckte Markup hingegen ermöglichten den Editoren nach einer kurzen Eingewöhnungsphase, auf übersichtlichere Art und Weise, den genetischen Markup in die Dokumente einzufügen. Die so annotierten Dokumente wurden nach ihrer Fertigstellung mit Hilfe von XSLT von einem Großteil des Word-spezifischen Markups befreit und nur die gewünschten Word-eigenen Formatierungselemente (Fußnoten, Tabellen, Listen und andere Formatierungen) in ein geeignetes XML-Format übernommen. Der überwiegende Teil eines Word-ML-Dokumentes besteht aus Markup, das spezifische formattechnische Verarbeitungsanweisungen enthält, die der systeminternen Darstellung und Verwaltung dienen und die für die Kodierung der Nuntiaturreportagen überflüssig waren. Darüber hinaus war das Format aufgrund seiner komplexen Struktur sehr speicherintensiv und deswegen nicht geeignet, in XML-basierten Datenbanken, wie sie in diesem Projekt zum Einsatz kamen, gespeichert zu werden. In Abbildung 2.10 ist ein Ausschnitt des farbigen ‘genetischen Markups’ eines Berichtsentwurfes in Word zu sehen.

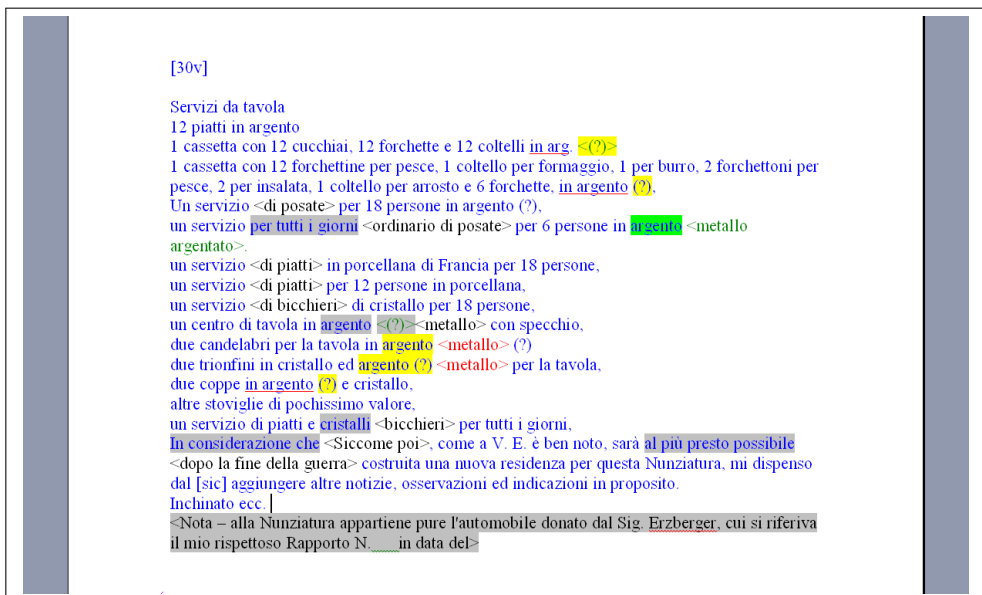


Abbildung 2.10 Bearbeitung in Word

³²³ Unter anderem war an der Aufnahme der Daten ein Team aus freiwilligen Seniorenstudenten beteiligt, das über ausreichende Kenntnisse im Bereich herkömmlicher Textverarbeitungsprogramme verfügte, darüber hinaus aber nicht für die Verwendung spezieller XML-verarbeitender Softwaresysteme geschult werden konnte.

Die Erstfassung des Textes wurde in blauer Schrift eingegeben und zeigt an, dass dieser Bericht von einem Stenotypisten aufgezeichnet wurde. Die übrigen farblich gekennzeichneten Textstellen signalisieren die Modifikationen des Textes durch die weiteren Bearbeiter (grün für den Uditor Lorenzo Schioppa, schwarz für Eugenio Pacelli, rot für den Privatsekretär Torricella, blau für unbekannte Stenotypisten). Einfügungen der einzelnen Personen wurden durch entsprechende Textfarben und schließende und öffnende spitze Klammern dargestellt. Die spitzen Klammern waren deswegen notwendig, da diese Einfügungen sonst nicht vom übrigen Text hätten unterschieden werden können, wenn ein Bearbeiter z.B. seinen eigenen Text korrigierte. Die Kodierung der Streichungen offenbart eine erste Schwäche dieser Art des Markups. Da Word farbige Streichungen von Text nicht unterstützte, mussten die Streichungsvorgänge etwas umständlich durch ein Highlighting-System ersetzt werden. Im Wesentlichen wurde versucht, die jeweilige Textfarbe des Bearbeiters auch für die Streichungssimulation zu verwenden. Diese Methode stieß allerdings dort an ihre Grenzen, wo der darunter liegende Text nur schlecht oder gar nicht mehr lesbar war, weil die verwendete Textfarbe als Highlight zu dunkel war. Für von Pacelli vorgenommene Streichungen wurde daher eine farblich graue Hinterlegung gewählt, für Schioppa ein Grünton, für die Stenotypisten blau und für den roten Text von Torricella gelb. Dies scheint auf den ersten Blick verwirrend, stellte aber in der alltäglichen Editionspraxis nach kurzer Eingewöhnungszeit kein Problem dar. Unterstreichungen konnten in der entsprechenden Farbe übernommen werden, da Word diese Funktionalität unterstützte. Die Reihenfolge, in der die einzelnen Bearbeitungsschichten aufeinander folgen, wurde in einem separaten XML-Element angegeben. In diesem Element wurden die Personennamen, die für die dazugehörigen Bearbeitungsschichten standen, durch Semikolon getrennt in ihrer korrespondierenden Textfarbe eingetragen. Dies war ein weiterer Nachteil, da bei einer erneuten Bearbeitungsschicht der selben Person nicht die gleiche Farbe verwendet werden konnte. Bei einer Bearbeitungsfolge von Pacelli-Schioppa-Pacelli etwa konnte zwar per Definition vereinbart werden, dass schwarzer Text (Pacelli), der nicht durch eine Streichung oder Einfügung gekennzeichnet war, in die erste Bearbeitungsschicht fiel und der Rest in die dritte, bei jeder weiteren Bearbeitungsschicht Pacellis funktionierte diese Vorgehensweise jedoch nicht mehr.

un centro di tavola in argento <(?)><metallo> con specchio,

Abbildung 2.11 Farbiges Markup

In Word-ML sind Textfarbe und Highlight im Attribut `w:val` der Elemente `w:color` und `w:highlight` gespeichert. Alle Textelemente `w:t` werden in `w:r`-Elemente (sogenannte runs) gekapselt. Diese `w:r`-Elemente können neben dem Text zusätzlich noch sogenannte property-Elemente (`w:rPr` - runnig-properties) enthalten, in die z.B. Formatierungseigenschaften, wie die Textfarbe und das Text-Highlighting, eingefügt werden können. Der Text „argento“ besitzt das Farb-Attribut `<w:color w:val="0000FF"/>` (blau) und ein Highlight-Attribut grau `<w:highlight w:val="light-gray"/>`. Die in Abbildung 2.11 gezeigte Textstelle wurde also von einem Stenotypisten aufgenommen

und anschließend durch die Streichung von Pacelli korrigiert. Das eingefügte grüne Fragezeichen <?>, das von Schioppa stammt (<w:color w:val="008000"/>), wurde ebenfalls durch Pacelli korrigiert. Aus dem XML-Format konnten mit Hilfe von XSLT die genetischen Informationen extrahiert und in folgendes vereinfachtes XML-Format transformiert werden:

```

1 <layer color="0000FF">un centro di tavola in </layer>
2 <layer color="0000FF" highlight="light-gray">argento</layer>
3 <layer color="008000" highlight="light-gray">&lt; (?)&gt;</layer>
4 <layer color="000000">&lt; metallo&gt;</layer>
5 <layer color="0000FF"> con specchio, </layer>

```

Listing 2.1

Mit jenem Format war es nun möglich, unter Zuhilfenahme von javascript eine dynamische Ansicht der unterschiedlichen Textschichten zu implementieren. Für Folio 30v aus Abbildung 2.10 sind die sequentiell einblendbaren Textschichten in Abbildung 2.12 zu sehen. Aus der Arbeit an dem Pacelli-Projekt ergaben sich einige interessante Fragestellungen. Was ließ sich aus dieser Art des Edierens textgenetischer Prozesse lernen? Welche Komponenten des Projekts eigneten sich für eine allgemeinere Beschreibung solcher Prozesse? Wo ließen sich Bearbeitungsmethoden und Darstellungsfunktionen verbessern oder erweitern? Wo lagen die Schwächen bei der Datenerfassung, Kodierung, Speicherung und Darstellung? Die Schlussfolgerungen werden in der folgenden Liste zusammengefasst.

Vorteile

1. Verbindung aller Vorteile eines Texteditors mit projektbezogenen Editionsanforderungen. Funktionalität vor allem in der Textverarbeitung musste nicht neu implementiert werden. Formateigenschaften (bold, italic, underline etc.), Fußnoten, Tabellen, Listen etc. ließen sich dank XML-Notation leicht aus dem Code extrahieren.
2. Intuitives Arbeiten durch wysiwyg-Prinzip und Verwendung von Farben zur Visualisierung von dynamischen Prozessen war möglich.

Nachteile

1. Bindung an kommerzielle Software (Microsoft Word).
2. Keine direkten Verlinkungsmöglichkeiten der Texte mit dem Digitalisat.

³²⁴ Dokument Nr. 4209, Nuntiaturreport, Entwurf, Pacelli an Gasparri, 1917-10-30, in: 'Kritische Online-Edition der Nuntiaturreport Eugenio Pacellis (1917-1929)', url: www.pacelli-edition.de/Dokument/4209

<p>30v</p> <p>Servizi da tavola 12 piatti in argento 1 cassetta con 12 cucchiali, 12 forchette e 12 coltelli in arg. 1 cassetta con 12 forchettine per pesce, 1 coltello per formaggio, 1 per burro, 2 forchettoni per pesce, 2 per insalata, 1 coltello per arrosto e 6 forchette, in argento (?), Un servizio per 18 persone in argento (?), un servizio per tutti i giorni per 6 persone in argento un servizio in porcellana di Francia per 18 persone, un servizio per 12 persone in porcellana, un servizio di cristallo per 18 persone, un centro di tavola in argento con specchio, due candelabri per la tavola in argento (?) due trionfini in cristallo ed argento (?) per la tavola, due coppe in argento (?) e cristallo, altre stoviglie di pochissimo valore, un servizio di piatti e cristalli per tutti i giorni, In considerazione che , come a V. E. è ben noto, sarà al più presto possibile costruita una nuova residenza per questa Nunziatura, mi dispenso dal [sic] aggiungere altre notizie, osservazioni ed indicazioni in proposito. Inclinato ecc.</p>	<p>30v</p> <p>Servizi da tavola 12 piatti in argento 1 cassetta con 12 cucchiali, 12 forchette e 12 coltelli in arg. <(?)> 1 cassetta con 12 forchettine per pesce, 1 coltello per formaggio, 1 per burro, 2 forchettoni per pesce, 2 per insalata, 1 coltello per arrosto e 6 forchette, in argento (?), Un servizio per 18 persone in argento (?), un servizio per tutti i giorni per 6 persone in argente <metallo argentato>. un servizio in porcellana di Francia per 18 persone, un servizio per 12 persone in porcellana, un servizio di cristallo per 18 persone, un centro di tavola in argento <(?)> con specchio, due candelabri per la tavola in argento (?) due trionfini in cristallo ed argento (?) per la tavola, due coppe in argento (?) e cristallo, altre stoviglie di pochissimo valore, un servizio di piatti e cristalli per tutti i giorni, In considerazione che , come a V. E. è ben noto, sarà al più presto possibile costruita una nuova residenza per questa Nunziatura, mi dispenso dal [sic] aggiungere altre notizie, osservazioni ed indicazioni in proposito. Inclinato ecc.</p>
<p>30v</p> <p>Servizi da tavola 12 piatti in argento 1 cassetta con 12 cucchiali, 12 forchette e 12 coltelli in arg. <(?)> 1 cassetta con 12 forchettine per pesce, 1 coltello per formaggio, 1 per burro, 2 forchettoni per pesce, 2 per insalata, 1 coltello per arrosto e 6 forchette, in argento (?), Un servizio <di posate> per 18 persone in argento (?), un servizio per tutti i giorni <ordinario di posate> per 6 persone in argente <metallo argentato>. un servizio <di piatti> in porcellana di Francia per 18 persone, un servizio <di piatti> per 12 persone in porcellana, un servizio <di bicchieri> di cristallo per 18 persone, un centro di tavola in argente <(?)> <metallo> con specchio, due candelabri per la tavola in argento (?) due trionfini in cristallo ed argento (?) per la tavola, due coppe in argento (?) e cristallo, altre stoviglie di pochissimo valore, un servizio di piatti e cristalli <bicchieri> per tutti i giorni, In considerazione che <Siccome poi>, come a V. E. è ben noto, sarà al più presto possibile <dopo la fine della guerra> costruita una nuova residenza per questa Nunziatura, mi dispenso dal [sic] aggiungere altre notizie, osservazioni ed indicazioni in proposito. Inclinato ecc. <Nota alla Nunziatura appartiene pure l'automobile donato dal Sig. Erzberger, cui si riferiva il mio rispettosso Rapporto N. in data del></p>	<p>30v</p> <p>Servizi da tavola 12 piatti in argento 1 cassetta con 12 cucchiali, 12 forchette e 12 coltelli <u>in arg.</u> <(?)> 1 cassetta con 12 forchettine per pesce, 1 coltello per formaggio, 1 per burro, 2 forchettoni per pesce, 2 per insalata, 1 coltello per arrosto e 6 forchette, <u>in</u> <u>argente</u> (?), Un servizio <di posate> per 18 persone in argento (?), un servizio per tutti i giorni <ordinario di posate> per 6 persone in argente <metallo argentato>. un servizio <di piatti> in porcellana di Francia per 18 persone, un servizio <di piatti> per 12 persone in porcellana, un servizio <di bicchieri> di cristallo per 18 persone, un centro di tavola in argente <(?)> <metallo> con specchio, due candelabri per la tavola in argente <metallo> (?) due trionfini in cristallo ed argente <(?)> <metallo> per la tavola, due coppe <u>in argente</u> (?) e cristallo, altre stoviglie di pochissimo valore, un servizio di piatti e cristalli <bicchieri> per tutti i giorni, In considerazione che <Siccome poi>, come a V. E. è ben noto, sarà al più presto possibile <dopo la fine della guerra> costruita una nuova residenza per questa Nunziatura, mi dispenso dal [sic] aggiungere altre notizie, osservazioni ed indicazioni in proposito. Inclinato ecc. <Nota alla Nunziatura appartiene pure l'automobile donato dal Sig. Erzberger, cui si riferiva il mio rispettosso Rapporto N. in data del></p>

Abbildung 2.12 Einblendbare Genese-Layer - 1. Stufe (oben links): Stenotypist (blau), 2. Stufe (oben rechts): Schioppa (grün), 3. Stufe (unten links): Pacelli (schwarz), 4. Stufe (unten rechts): Torricella (rot)³²⁴

3. Keine Offenheit für andere Editionstypen.
4. Schlechte Kodierung der Editionsprozesse - Vermischung von Form und Inhalt durch Kodierung der einzelnen Layer mit Farb-Attributen.
5. Keine ausreichende Sequenzierung der einzelnen Bearbeitungsschritte.
6. Keine Unterstützung anderer dynamischer Prozesse (Variantenverzeichnung, Synoptische Textdarstellung auf Zeilenebene etc.).
7. Eine einmal für einen bestimmten Editor (Word) programmierte Funktionalität (Extrahieren von Formateigenschaften, Fußnoten, Tabellen etc.) ließ sich nur mit großem Aufwand auf andere Editoren übertragen.

2.4 Texttopographie

Ein zentrales Problem aktueller digitaler textgenetischer Editionen besteht in der unzureichenden Kodierung topographischer Informationen und den daraus resultierenden beschränkten Möglichkeiten der Analyse und der Visualisierung. Gute Gründe für eine genaue Verzeichnung topographischer Informationen gibt es genug.³²⁵ Aufgabe der Transkription ist es, eine möglichst genaue Beschreibung des Originals zu gewährleisten. Da eine solche Grundtranskription idealerweise die Basis aller weiteren Verzeichnungsebenen darstellt, sollte diese unabhängig von einem bestimmten Erkenntnisinteresse so viele Informationen wie möglich enthalten. Diese Informationen sollten sich nach Möglichkeit objektiv an den physischen Gegebenheiten des Originals orientieren. Da jede Informationskodierung schon ein Prozess der Deutung ist, sollte diese zumindest eindeutig sein. Ein gutes Beispiel für eine mehrdeutige Informationskodierung bietet das place-Attribut der TEI.³²⁶ Die Angaben von positionellen Informationen werden hier auf einer sehr hohen Abstraktionsebene kodiert. Für eine Einfügung ist es z.B. gängige Praxis, innerhalb des add-Elements mit dem place-Attribut zu kennzeichnen, wo die Einfügung im Original positioniert wurde. Als Werte werden dort unter anderem vorgeschlagen: below (below the line), bottom (at the foot of the page), margin (in the margin: left, right, or both), inline (in a predefined space, for example left by an earlier scribe), etc. Eine solche Kodierung ist natürlich alles andere als exakt und eindeutig. Eine relative Angabe der Positionierung wie ober-, unter- oder innerhalb eines Referenzobjektes ohne die genaue Angabe 'wo denn genau oberhalb' etc., also ohne die genauen Koordinaten, sagt nur wenig über die tatsächliche Position aus.³²⁷ Insgesamt schenkt die TEI der genauen physischen Dokumentstruktur nur wenig Aufmerksamkeit. Sie übernimmt solche

³²⁵ Vgl. u.a. Dirk van HULLE: 'Erronymous' Intentions from Joyce to Danielewski, in: *Variants – The Journal of the European Society for Textual Scholarship* 1 2002, S. 123–141 und Hans Walter GABLER: For Ulysses – A Once and a Future Edition, in: *Variants – The Journal of the European Society for Textual Scholarship* 1 2002, S. 85–105.

³²⁶ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/ref-att.placement.html>

³²⁷ Vgl. auch Gerrit BRÜNING/Katrin HENZEL/Dietmar PRAVIDA: Multiple Encoding in Genetic Editions: The Case of 'Faust', in: *Journal of the Text Encoding Initiative* 2013, URL: <http://jtei.revues.org/697>, §15.

topographischen Taxonomien aus Fachdisziplinen, in denen genauere räumliche Kodierungen eine untergeordnete Rolle spielen, da es dort weniger um die physischen, sondern mehr um die logischen Textstrukturen geht. Allerdings ist dies nicht ganz 'ungefährlich', da die logische Ebene nicht ohne weiteres von der physischen zu trennen ist. Oft lassen sich erst aus der genauen physischen Transkription Rückschlüsse auf logische Strukturen ziehen. Das Layout eines Textes enthält durch die Formatierung und die Positionierung der Textsegmente wichtige Anhaltspunkte dazu, wie ein Text zu lesen ist, aber auch dazu, wie er entstanden ist. Schreibräume können auf unterschiedliche nicht-lineare Weise in Zusammenhang stehen,³²⁸ z.B. in unterschiedlichen chronologischen Zusammenhängen. Neben der physischen Abfolge, also der linearen Verteilung der Zeichen auf dem Textträger, muss unterschieden werden zwischen einer 'tatsächlichen' Chronologie (in welcher Abfolge sind die Zeichen und Zeichenketten notiert worden) und einer logischen Chronologie (wie sind diese in eine Abfolge zu bringen, damit ein 'sinnvoller' Text entsteht). Da man sich hier auf dem Gebiet der Deutung von Text bewegt, sind natürlich auch alternative Lesarten von Chronologien zu berücksichtigen (siehe dazu Abschnitt 2.5). Um diese Lesarten für den Benutzer nachvollziehbar zu machen, ist es geradezu unumgänglich, ihm die genauen Positionen der Schreibräume mitzuteilen bzw. diese auch im Digitalisat sichtbar zu machen, da die Topographie der Schreibräume wichtige Indizien liefern kann, wie ein Text zu lesen ist. Nur anhand solcher Indizien kann sich der Benutzer selbst ein Bild davon machen und gegebenenfalls eigene Interpretationen über den Zusammenhang der Schreibräume aufstellen. Was hier also versucht wird, ist, nicht-lineare Schreibräume zu linearisieren und zwar zumeist nur in einer Form, entweder durch die lineare Anordnung in einer vermeintlichen Abfolge des tatsächlichen Schreibprozesses oder in einer rekonstruierten Abfolge, die angibt, in welcher Abfolge diese Schreibräume zu lesen sind. Die Linearität der Transkription in diesen Darstellungsformen ist nicht in der Lage, die eigentliche Räumlichkeit und ihre positionelle Anordnung wiederzugeben: „Diese Schrift-Stücke immer schon in den linearen Fluss des konstruierten Textes einzubinden, heißt die Deutung zu codieren und ihre Grundlagen auszublenden.“³²⁹

Die Linearisierung nicht-linearer Prozesse ist auch eines der Hauptprobleme aktueller digitaler textgenetischer Editionen. Schaut man sich ein Beispiel aus dem Wittgensteinarchiv oder aus der Musil-Edition an, wird dies schnell deutlich (Abbildung 2.13). Hier wird zum einen versucht, genetische Informationen in linearisierter lesbarer Form darzustellen, und zugleich wird angestrebt, innerhalb dieser Textdarstellung möglichst viele Informationen der ursprünglichen, meist nicht linearen Textgestalt unterzubringen. Hier liegt das eigentliche Problem darin, dass versucht wird, durch die zur Verfügung stehenden formatierungstechnischen Möglichkeiten der jeweils verwendeten Software oder der in browserbasierten Anwendungen verwendeten Markup-Sprache (HTML) und dem zur Verfügung stehenden Zeichenvorrat, einigermaßen einleuchtende Symbolisierungen für verschiedene Arten von Textphänomenen zu verwenden. Dabei werden munter alle zur Verfügung stehenden Möglichkeiten genutzt: Textfar-

³²⁸ Vgl. dazu auch Sahle: „Dokument-Texte sind zunächst nicht Ketten von Zeichen, sondern bestenfalls Ketten von Schreibräumen“ SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 296.

³²⁹ Ebd., S. 297.

dieser Stelle der Lesefluss unterbrochen wird, da mit dieser Kodierungsvariante der Text erst nach der Einfügung fortgesetzt wird und sich die Zeile so auf unnatürliche Weise verlängert.³³³ Hier muss man sich fragen, ob es wirklich sinnvoll ist, zugunsten der simulierten Anzeige einer interlinearen Einfügung auf andere Informationen der Textgestalt (Zeilenlänge etwa) zu verzichten.³³⁴ Dieses Verfahren birgt auch große Probleme bei der Kodierung von Einfügungen, die aus wesentlich längeren Textpassagen bestehen, z.B. wenn der Text als interlineare Einfügung beginnt, sich aber am Textrand weiter fortsetzt. Die Formatierung in grau ohne Hochstellung (Musil) beinhaltet im Prinzip die gleichen Risiken. Es würde auch zu weiteren Problemen führen, wenn ein hochgestellter Text wiederum interlineare Einfügungen enthält. Nach dieser Methode müsste eine Hochstellung über der Hochstellung oder eine graue Hinterlegung auf der grauen Hinterlegung realisiert werden. Dieses Problem kann wiederum nur durch weitere visuell unterscheidbare Formatierungen (z.B. ein dunkleres grau) gelöst werden oder durch ein kaskadierendes System von Hochstellungen, so wie Saller es vorschlägt.³³⁵ Ein anderes Problem besteht darin, dass diese einmal so verwendeten Formatierungen in den Editionsrichtlinien festgelegt sind und auch nur für diesen Sachverhalt verwendet werden können. Sollte z.B. eine Textpassage im Original wirklich mit einer bestimmten Farbe oder einem Grauton (wie bei Musil) hinterlegt worden sein, kann diese Formatierung eigentlich nicht mehr dafür verwendet werden oder nur dann, wenn sie in diesem speziellen Fall zusätzlich erläutert wird (mit einer Fußnote oder einem erklärenden Tooltip z.B.). Die Problematik an einer solchen Art der Visualisierung von topographischen und anderen textgenetischen Informationen liegt in der Verwendung von Formatierungen für Zwecke, für die sie nicht bestimmt sind. Das Hochstellen des Textes in HTML mag sinnvoll sein, um in einem linearen Text z.B. eine hochgestellte Fußnote zu positionieren. Hiermit wird dann angezeigt, dass dieses Zeichen oder dieser Text nicht zum eigentlichen Fließtext gehört. In Bezug auf die textgenetische Positionierung sagt diese Formatierung aber nichts darüber aus, wo genau diese Hinzufügung positioniert ist, wann sie eingefügt wurde oder von wem. Für diese Informationen müssten weitere Formatierungen verwendet werden.³³⁶

Im Prinzip handelt es sich bei dieser Art der Informationsvisualisierung um die gleiche, schon in Abschnitt 1.2.2 beschriebene, kritisch zu betrachtende Praxis der Vermischung von Inhalts- und Ausdrucksseite. Hier wird ähnlich wie bei der semantischen Auszeichnung einer Kapitelüberschrift durch ein `<h>` Element eine beliebige Formatierung gewählt, um den semantischen Gehalt des Einfügens oberhalb einer Zeile zu kodieren. Freilich passiert dies in der Regel nur auf der Ausgabeseite, und

schaft 2009, hrsg. v. Gertraud MITTERAUER, S. 283–290, hier S. 285. Liste der verwendeten diakritischen Zeichen: <http://www.uibk.ac.at/germanistik/stifter/witiko/stemma/>

³³³ Andere Methoden, dies mit den Möglichkeiten von HTML zu lösen, werden im nächsten Kapitel beschrieben.

³³⁴ In dem Musil-Beispiel wird deswegen auch zugunsten einer kürzeren Zeilendarstellung auf die exakten Zeilenumbrüche verzichtet, und somit entspricht die Zeilennummerierung im Beispiel nicht mehr der des Originals.

³³⁵ Vgl. SALLER: HNML (wie Anm. 316). Das kaskadierende System verschärft allerdings das Problem der überlangen Zeilen.

³³⁶ Z.B. unterschiedliche Schrifttypen oder Farben, anhand derer man den Schreiber identifizieren kann.

die eigentliche Grundkodierung wird durch eine eindeutige formatierungsunabhängige Auszeichnungsstruktur bereitgestellt.³³⁷ Allerdings besteht keine saubere Trennung zwischen den beiden Ebenen der Ausdruckseite in Form einer genauen 'diplomatischen' Darstellung (dem HTML-formatierten Text) und der tatsächlichen 'realen' Inhaltsseite, dem Original (oder dem Digitalisat des Originals). Jegliche symbolhafte Art der Informationskodierung, die nicht den genauen physischen Beschaffenheiten des Originals entspricht, führen bei der Rezeption des Textes zwangsläufig zu Informationsverlust,³³⁸ da genügt es auch nicht, das Faksimile neben die Transkription zu stellen und zu argumentieren, dass so ja nun alle Informationen vorhanden wären. „Alle Informationen der vorliegenden Dokumente sollen – im Idealfall – in der Transkription erhalten bleiben.“³³⁹ Das trifft bei der Kodierung in den genannten Beispielen auch zu, aber welcher Mehrwert ergibt sich daraus, wenn die Visualisierung in sogenannten 'diplomatischen Darstellungen' nicht zu einem genauem Abbild des Originals führt, sondern nur eine reichlich ungenaue symbolhafte Annäherung dessen ist, was wirklich die Materialität des Originals ausmacht, also nur „Simulation von Layout durch Zeichen“³⁴⁰ oder unzureichenden Formatierungen ist? Wenn eine diplomatisch genaue Darstellung angestrebt wird, sollte diese auch wirklich exakt diese Informationen beinhalten. Dass sie angestrebt wird steht außer Frage, versucht ja z.B. die Musil-Edition die Randbemerkungen auch dort zu positionieren, wo sie im Original zu finden sind. Dann stellt sich aber die Frage, warum nur so inkonsequent? Die Beantwortung fällt hier nicht schwer, sie ergibt sich aus den zuvor schon beschriebenen Restriktionen des verwendeten Formats und den eingeschränkten Visualisierungsmöglichkeiten der jeweiligen Software. Nicht nur die räumliche Anordnung oberhalb oder unterhalb von Textelementen führt hier zu Problemen. Auch das Positionieren und Visualisieren von Unterstreichungen und Streichungen und anderen graphischen Elementen unterliegt starken Limitierungen in HTML. So ist eine Streichung mit der css-Eigenschaft `text-decoration:line-through` nur mittig auf der Zeile positionierbar, es ist nicht möglich, die Dicke der Streichung zu beeinflussen oder der Streichung ein weiteres Formatierungsattribut hinzuzufügen, wie etwa `dotted` für eine punktierte Streichung. Bei den `underline`-Elementen verhält es sich ähnlich. Hier kann nur mit Hilfe eines Tricks eine Unterstreichung mit weiteren Darstellungsoptionen versehen werden. Ein `text-decoration:underline` mit der css-Eigenschaft `border-bottom:1px dotted;` (oder auch `dashed` für gestrichelte Linien) erlaubt es zwar, gewisse visuelle Eigenschaften zu simulieren, weitere Formatierungen werden aber auch hier nicht unterstützt. Dies wäre allerdings sinnvoll. Es kann z.B. vorkommen, dass ein Autor verschiedene Arten von Streichungen und Unterstreichungen verwendet, die semantisch unterschiedliche Informationen beinhalten. Eine variierende Stärke oder Vehemenz der Durchstreichung kann z.B. Aussagen darüber zulassen, wie 'unzufrieden' der Autor mit dem Einfügen der jeweiligen Textpassage war. Gegebenenfalls wird er solche Textpassagen auch mehrfach gestrichen haben, was soweit gehen kann, dass der Text komplett getilgt

³³⁷ Und so sollte es ja nach gängiger Forschungsmeinung auch sein: layoutneutral Kodieren und für den jeweiligen Ausgabekontext formatieren.

³³⁸ Vgl. RENEAR: Out of Praxis (wie Anm. 68), 108ff.

³³⁹ SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 254.

³⁴⁰ Ebd., S. 185.

wurde und nicht mehr lesbar ist. Andererseits kann eine zaghaft (oder gestrichelt) ausgeführte Streichung ein Signal für ein gewisses Zögern sein oder eine Ungewissheit, ob diese Textstelle wirklich gestrichen werden soll. Es ist auch denkbar, dass der unterschiedlichen Verwendung von Unterstreichungen eine eigene Systematik zugrunde liegt, dass farbige, punktierte, mit verschiedenen Schreibutensilien ausgeführte Unter- oder Durchstreichungen unterschiedliche Bedeutungen haben. Dies alles lässt sich mit den Mitteln von HTML nicht adäquat darstellen.

Stellt man neben eine Transkription, wie sie in dem Musil- oder Wittgenstein-Beispiel zu sehen sind, das zugehörige Faksimile, ist man nicht selten überrascht, wie wenig dieses mit der zumeist 'diplomatisch' oder auch 'ultradiplomatisch' genannten Transkription zu tun hat. Die eigentliche Aufgabe der Transkription sollte ja sein, ein möglichst genaues Abbild des Originals zu gewährleisten.³⁴¹ Und dies beinhaltet dann auch die möglichst originalgetreue Abbildung der Topographie und anderer graphischer Symbole.³⁴² Das Bild auf dem Faksimile stellt sich in der Regel aber ganz anders dar. Trotz des oftmals beigefügten Faksimile ist zudem eine Orientierung im Original anhand des genetischen Markups des 'Lesetextes' nur sehr schwer möglich. Es werden auch nur wenige bis gar keine Verlinkungsfunktionen angeboten, um sich eine bestimmte Textstelle im Faksimile direkt anzeigen lassen zu können. Insgesamt ist diese Art der Visualisierung eher als Versuch zu betrachten, die klassisch gedruckten statischen diplomatischen Abbildungen in digitaler Form nachzubilden.

2.4.1 Diplomatische Ansichten

„Der edierte Text tritt nicht mehr verdrängend an die Stelle der Dokumente, sondern erklärend. Umgekehrt ist das Faksimile nicht nur Illustration des 'eigentlichen Textes', sondern liefert Informationen zum Verständnis der Genese und zeitgenössischen visuellen Aussage“³⁴³ Dabei ist es nicht so, dass das digitale Faksimile die diplomatische Transkription ersetzen würde. Es wird zwar immer noch häufig bemerkt, dass zu einer Überprüfung am Original die Hinzugabe eines Faksimiles ausreichen müsste und man salviert diese Entscheidung häufig mit der Unmöglichkeit, die ursprüngliche Gestalt des Textes nicht imitieren zu können,³⁴⁴ aber wie Buzzetti schon treffend bemerkt, kann das Faksimile keinen ausreichenden Ersatz bieten: „The image of a document, for example, does not make its diplomatic transcription superfluous. The graphic information conveyed by an image is not the textual information conveyed by a transcription.“³⁴⁵ Auf den ersten Blick gut organisiert birgt dies einige Risiken. Bei diesem Beispiel handelt es sich um ein Tabellenlayout, das 'unnatürliche' Segmentierungen verwendet. Der Text wird in Spalten- und Zeilenelemente verteilt. Dabei

³⁴¹ Vgl. PICHLER: Transcriptions (wie Anm. 31), S. 690.

³⁴² KRAFT: Editionsphilologie (wie Anm. 273), 114ff.

³⁴³ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 270.

³⁴⁴ „Da die ursprüngliche Form des Dokuments häufig nicht gänzlich imitiert werden konnte, besteht die Möglichkeit, sich ein Faksimile jedes Dokuments anzeigen zu lassen.“ Noah BUBENHOFER u. a.: XML-Technologien als Grundlage dynamischer Textpräsentation. Die digitale Quellenedition Der Zürcher Sommer 1968, in: Jahrbuch für Computerphilologie 9 2009, URL: <http://computerphilologie.digital-humanities.de/jg07/bubenh.html>, §20.

³⁴⁵ BUZZETTI: Diacritical ambiguity (wie Anm. 75), S. 176.

werden die Randnotizen in Zellen um den Haupttext organisiert (Abbildung 2.14 roter Rahmen).

I, 1 : L'entrée de Charles au collège - Brouillons, vol. 1, folio 3v	
<p>ce fut un vacarme suprême. en comme un flot débordant un hourr une chute d'eau subite un hourrah s'élança - on criait - on miaulait - on gloussait on répétait charbovari, charbovari - on se tordait sur les bancs - on trépanait des pieds - puis & les rires d'abord mêlés & formant masse se distinguèrent en s'abaissant - puis peu à peu ils rentrèrent - contenus - isolés com par-frais dans les angles & sonores contenus comme une bande d'oiseaux lâchés dans un appartement et dont on entend vitres sonner sur les carreaux, les coups d'ailes - qui fait d'abord un gd froufrou et dont on entend ça & là dans les angles sonner - heurte son vol aux meubles va taper du bec contre les & heurte aux murs son vol étouffé vitres - puis qui s'apaise peut doucement petit à petit & rentre dans la main - à m doucement à mesure qu'on les rattrape & petit à petit</p>	<p>alors, comme une bande d'ois - alors - ce fut un vacarme suprême - les rires ... (du rire) et à la fin comme sonnaient encore isolément comme une bande d'oiseaux lâchés dans un appartement, qui vont battre de l'aile contre les vitres - et ça meurt - on finit par distinguer son nom Charles Bovary encore et comme il n'a pas de livres, on lui dit de venir s'asseoir sur le petit banc et de suivre. cherche il veut prendre sa casquette - qu'un écolier a lâché lancé sous un le banc - que voulez-vous - ma casq. on allait 500 v. à toute la classe - l'ordre se rétablit - par çà par là qq boulettes de mie de pain lancées à la figure - qu'il essuie stoïquement avec son mouchoir à la récréation de quatre heures on l'entoure - un peu d'étonnement (costume) et on apprend ses antécédents - prquoi venu à cette époq. fuir* [illis] ce n'était pas sans sacrifices que les parens l'avaient mis au collège - ce qu'étaient ses (portraits) parens - vie de Charles chez eux - il eut à souffrir qq temps au collège - ee qu'il y fut - isolement - correspondant - esprit - caractère - vie - sort à la fin de la 3e*. il se mit à étudier la médecine liés* au collège)</p>

Abbildung 2.14 Diplomatische Transkription (Tabellenstruktur)

Andersherum ist es allerdings auch richtig, dass die diplomatische Transkription das Original oder das Faksimile nicht vollständig ersetzen kann, besonders dann nicht, wenn es sich um eine Art normalisierte diplomatische Transkription handelt, die, wie im Beispiel von Wittgenstein, zwar viele diplomatische Informationen enthält, aber beispielsweise nicht die genauen topographischen Koordinaten des Originals. Der Sinn eines Textes erklärt sich nämlich nicht nur aus einem alphanumerischen Code, sondern auch aus visuellen Merkmalen.³⁴⁶ Bei Zeitungen beispielsweise spielen das Layout, die

³⁴⁶ Hierzu Sahle: „nicht immer [ist] davon auszugehen [...], der Sinn von Texten würde vom Leser ausschließlich über den Bestand an Alphabetzeichen oder Wörtern hergestellt. Vielmehr dienen gerade die vielfältigen visuellen Merkmale von Schrift und Layout als grundlegende 'Verarbeitungsanweisungen', die bereits vor dem Entziffern der Buchstaben vorgeben, wie (und als was) ein Text zu lesen ist.“ SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 159.

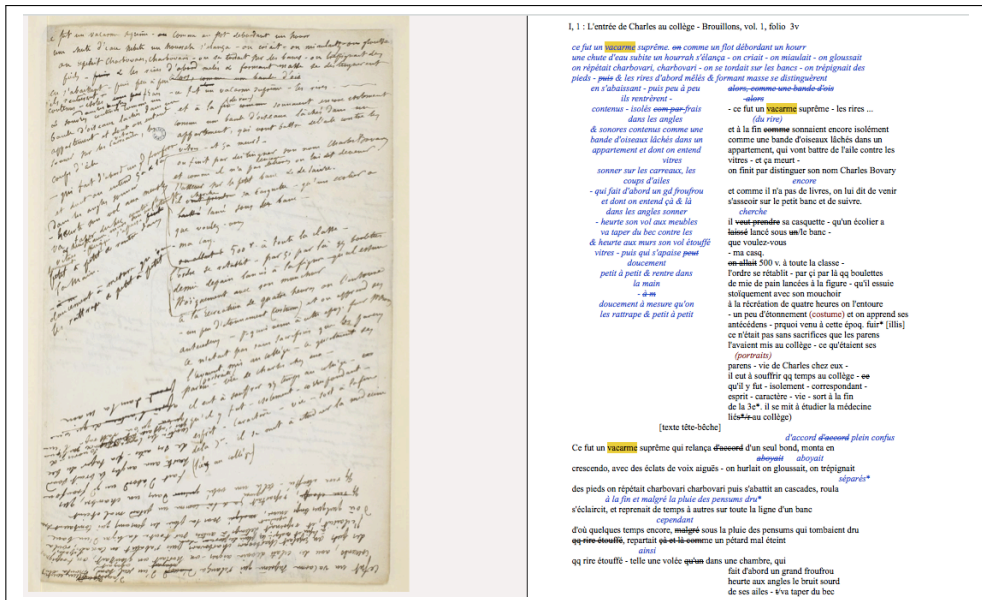


Abbildung 2.15 Diplomatische Transkription (Digitale Variante)

Überschriften und Textgrößen eine zentrale Rolle bei der Interpretation der Information. So lässt sich schon anhand der Verteilung und der Größe der Artikel erkennen, welche Bedeutung einer bestimmten Nachricht beigemessen wurde. Auch im Bereich der Lesehilfe spielen die topographischen Informationen eine nicht zu unterschätzende Rolle. Eine schwer lesbare oder durch Streichung fast unkenntlich gemachte Textpassage kann durch eine geeignete topographisch genaue Darstellung die Lesbarkeit erleichtern.

Wenn diese Informationen in Webbrowsern visualisiert werden, wie in Abbildung 2.15 zu sehen,³⁴⁷ geschieht dies in der Regel durch HTML-Kodierungen, die das Dokumentlayout entweder in Tabellenform organisieren oder die einzelnen Textobjekte mit `div`-Elementen positionieren. Die Positionierung innerhalb der Zellen geschieht mit den `align`-Attributen `left`, `center` und `right`. Genaue Textpositionen, die nicht durch diese Attribute abgedeckt werden, können nur durch eine Reihe führender gesperrter Leerzeichen (` `) simuliert werden, so dass es zu umständlichen Hilfskonstrukten wie folgenden kommt (Abbildung 2.14 kleiner roter Rahmen im Haupttext):

Charles Bovary

[illegible]

Diese Kodierung ist sehr ungenau, da hier (wahrscheinlich) für jedes Schriftzeichen ab Beginn der Zeile bis zu dem Punkt der interlinearen Einfügung ein Leerzeichen eingefügt wird. In HTML-Schriften haben aber die einzelnen Zeichen unterschiedliche

³⁴⁷ http://www.bovary.fr/folio_visu.php?folio=1591&mode=sequence&mot=vacarme

Rechtecke pixelgenau an die gewünschte Position verschieben. Mit der copy-Taste werden diese Elemente mit Hilfe der cloneNode-Funktion von javascript dupliziert und neben die Dokumentansicht mit absoluten Koordinaten positioniert. Gleichzeitig wird in die div-Elemente der Text (in diesem Falle ein Platzhaltertext) eingefügt. Somit ließen sich auf sehr einfache Weise auch komplexere Dokumente, die nicht einem einfachen Tabellenlayout entsprechen, mit Textsegmentierungen versehen. Die Nachteile sind aber auch bei diesem Verfahren offensichtlich. Textblöcke müssen sich zwangsläufig überschneiden, um das entsprechende Layout des Originals zu simulieren. Dabei wird oft Text in einen Block eingeschlossen, der nicht in diesen gehört. Man könnte nun zwar diese Textblöcke fragmentieren und soweit verschachteln, bis nur der gewünschte Text in einem Textblock erscheint, allerdings würde dies einen zusätzlichen Aufwand bedeuten und die Fragmente müssten durch einen Verknüpfungsmechanismus verbunden werden, um ihre Zusammengehörigkeit zu kennzeichnen (Abbildung 2.17).

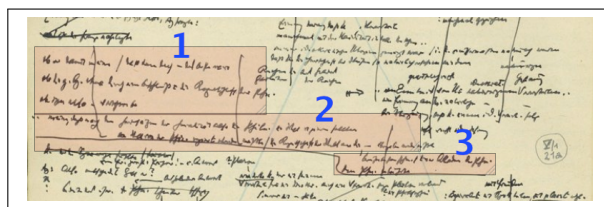


Abbildung 2.17 Segmentierung mit div-Elementen

Darüber hinaus gibt es weitere Probleme, die so nicht gelöst werden können. Neben der Positionierung des Textes sind z.B. auch unterschiedliche Textneigungen oder senkrecht positionierte Texte zu beachten (Abbildung 2.18).³⁴⁸

Hierfür müsste HTML in der Lage sein, Textrotationen in beliebige Richtungen auszuführen. Die Verlinkung zwischen dem Digitalisat und dem Text kann nur sehr ungenau über rechteckige Segmente realisiert werden. Bei sich neigendem Text kann so z.B. keine genaue Verlinkung auf eine Zeile des Dokumentes erzeugt werden. Unterschiedliche geometrische Formen, wie z.B. Polygone müssten eine genauere Markierung von nicht rechteckigen Textblöcken ermöglichen. Auch dies lässt sich nur unter großem Aufwand in HTML realisieren.

Neben dem genannten Grund, dass aktuell noch keine geeigneten standardisierten Markup-Konzepte bestehen, um topographische Informationen genauestens zu kodieren, spielt sicherlich auch die Tatsache eine Rolle, dass diese Art der Kodierung nur schwer manuell zu leisten ist, weil der Kodierungsaufwand immens wäre. Es mangelt bis heute an geeigneten Werkzeugen, um die topographischen Informationen mit dem Digitalisat auf eine Art und Weise zu verknüpfen, dass nicht nur die Referenz auf die Schreibräume gegeben ist, sondern dass sich anhand dieser Informationen der Text auch deckungsgleich über dem Digitalisat positionieren lässt. In letzter Konsequenz sollten diplomatische Transkriptionen die Möglichkeit bieten, textüberblendend in das Original eingefügt werden zu können. Nur so kann die Transkription wieder näher an das Original gebunden³⁴⁹ und eine genaue Überprüfbarkeit

³⁴⁸ Abbildung rechts: Amsterdam, Van Gogh Museum, inv. no. b5 V/1962. Entnommen aus: <http://www.vangoghletters.org/vg/letters/let005/letter.html>

³⁴⁹ "A good principle is to link any and all tagged text to the source of information. Just as print

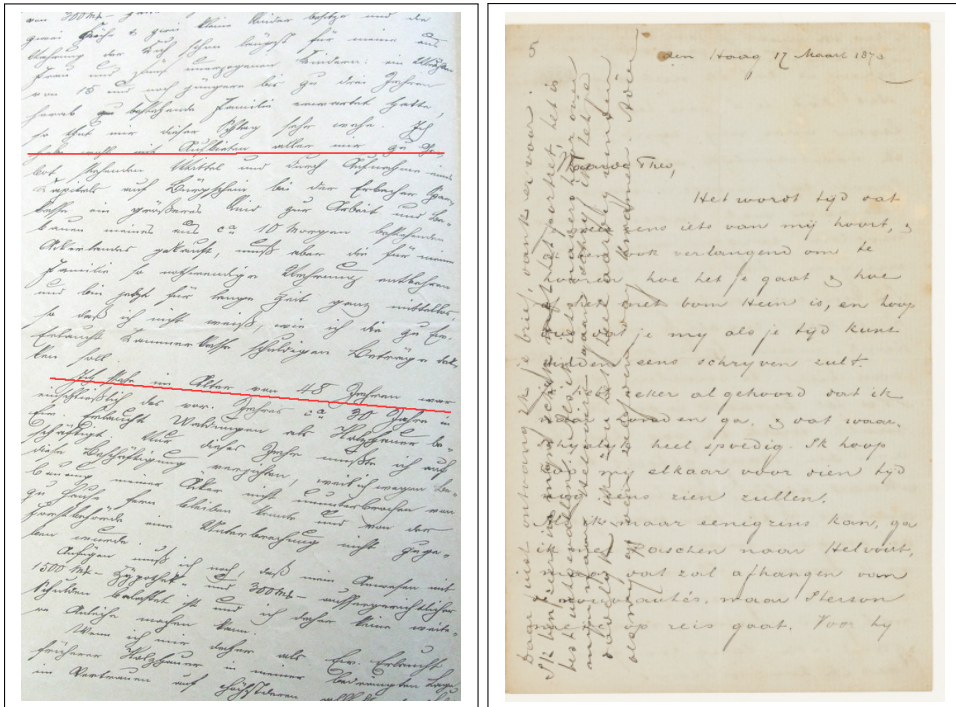


Abbildung 2.18 Textneigung

gewährleistet werden.³⁵⁰ Hierzu müssten also andere Formate als HTML verwendet werden. Vektorgrafikformate sind für solche Aufgaben weitaus besser geeignet. Unter <http://www.digitalvariants.org> kann eine in FLASH implementierte Demoversion aufgerufen werden, die zeigt, wie solche Textüberblendungen realisiert werden können.³⁵¹ Allerdings handelt sich hierbei nur um eine Studie zu diesem Thema, ein Annotationswerkzeug oder eine Beschreibung, wie solche Informationen zu kodieren sind, existieren bislang nicht. Im praktischen Teil der Arbeit in Abschnitt 3.3 wird beschrieben, wie ein solches Werkzeug in SVG implementiert werden kann.

editions gmove from manuscript to typescript, an image-based electronic edition must move from typescript back to manuscript.”KIERNAN: Digital Faksimiles (wie Anm. 281).

³⁵⁰ Vgl. Prätor: „Statt den Text von den Dokumenten abzulösen, soll[te] er aus ihnen abgeleitet werden [...] die Edition soll[te] so gestaltet werden, dass die Konstruktion des Textes aus den Dokumenten einleuchtend und nachvollziehbar wird.“ Klaus PRÄTOR: Ceci n'est pas un texte? Zur Rede über die Materialität von Texten – insbesondere in den Zeiten ihrer Digitalisierung. In: Materialität in der Editionswissenschaft 2010, hrsg. v. Martin SCHUBERT, S. 421–428, hier S. 428.

³⁵¹ www.digitalvariants.org/variants/valerio-magrelli. Vgl. auch: D. FIORMONTE/C. PUSCEDDU: The Text as Product and Process. History, Genesis, Experiments, in: Manuscript, Variant, Genese – Genesis 2006, hrsg. v. E. VANHOUTTE, S. 109–128.

2.5 Chronologien und Lesarten

„Text ist das Ergebnis eines Schreibprozesses. Transkription aber ist die Notation eines individuellen Leseprozesses. Notiert wird allerdings nicht nur das Beobachtete, sondern zugleich immer auch die Beobachtung. Egal welche Haltung zum Text wir einnehmen, immer konstruieren wir so neue Texte“³⁵²

Die Vorteile des Digitalen Mediums für die Darstellung dynamischer Prozesse sind schon früh in der Editionswissenschaft erkannt worden. Die Aussage von Gabler „Die Dynamisierungen und Interaktivierungen – und dabei nicht zuletzt die Versinnlichung – der Korrelation von Text und Dokument, welche das virtuelle Medium erlaubt, erfüllen im Kern nichts anderes als die Aufgaben der Apparate von Editionen im Medium von Schrift und Papier“³⁵³ bezieht sich auf die Möglichkeiten, den Textkritischen Apparat auflösen und die dort verzeichneten Informationen durch geeignete Techniken wieder näher an das Original binden zu können. Aber was heisst das genau für textgenetische Prozesse? Dass z.B. Einträge wie 'hds. gestrichen durch Schreiber XY' jetzt nicht mehr unter dem transkribierten Text innerhalb des Apparates verzeichnet sind, sondern direkt in diesem (mit Hilfe von Popup-Fenstern oder ähnlichen Techniken) angezeigt werden, bedeutet noch keine wirkliche Dynamisierung des Textes, zumindest nicht in dem Sinne, wie sie Lebrave für Textgenetische Editionen fordert. Nach Lebrave besteht die Textgenese aus einer doppelten Hypertextstruktur.³⁵⁴ Als wesentliche Eigenschaften identifiziert er für textgenetische Prozesse ihre „non-linearity, nonhierarchy, granularity, connectivity, and variability.“³⁵⁵ Die Natur von genetischen Schreibprozessen ist ihm zufolge geradezu exemplarisch geeignet für hypertextuelle Modelle. Für ihn handelt es sich dabei immer um eine doppelte Struktur. Er macht dies am Beispiel der Schreibprozesse von Blaise Pascal und Stendhal deutlich. Dort verortet er die erste Struktur in dem Prozess der Produktion (des geistigen Schaffensprozesses): „This private hypertext is inscribed in the space of production; it is part and parcel of the man who engraved it in his memory and partially transcribed it on paper.“³⁵⁶ und die zweite auf der Seite der Rezeption, in der Textvarianten und Fragmente durch Systeme sinnerschließender Verknüpfungen beschrieben werden sollten, um dem Leser mit Hilfe der Links die Möglichkeit zu geben, einen eigenen 'Pfad' durch diese Netzwerkstruktur zu verfolgen: „Reading becomes a dynamic activity in which the reader participates in the process of producing the object. Readers can create their own paths through a hypertext by choosing how to circulate in the network. They can also create an original 'hyperobject' inside the hypertext by exploiting the

³⁵² SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 274.

³⁵³ Hans Walter GABLER: Das wissenschaftliche Edieren als Funktion der Dokumente, in: Jahrbuch für Computerphilologie 8 2007, URL: <http://computerphilologie.digital-humanities.de/jg06/gabler.html>, §14.

³⁵⁴ Hier ist noch deutlich der Einfluss der Hypertextdiskussionen der 90er Jahre zu spüren. Vgl. dazu auch Jean-Luis LEBRAVE: Hypertext und textgenetische Edition, in: Beihefte zu Editio, Bd 10: Textgenetische Edition 1998, hrsg. v. Hans ZELLER, S. 329–345.

³⁵⁵ Jean-Louis LEBRAVE: Hypertexts - Memories - Writing, in: Genetic Criticism. Texts and Avant-textes 2004, hrsg. v. Jed DEPPMAN/Daniel FERRER/Michael GRODEN, S. 218–237, hier S. 222.

³⁵⁶ Ebd., S. 228.

property of granularity and creating their own links.“³⁵⁷ Auf der einen Seite bedeutet das, die chronologische Abfolge eines Schreibprozesses eines Autors sichtbar zu machen, und auf der anderen Seite, dem Benutzer die Möglichkeit zu geben, selbst unterschiedliche chronologische Pfade durch die Textgenese beschreiten zu können, also z.B. selbst zwischen unterschiedlichen Lesarten oder Varianten zu wählen ohne etwa einen Leittext vor Augen zu haben.

2.5.1 Chronologische Abfolgen auf Mikro- und Makroebene

Bei chronologischen Abfolgen von Schreibprozessen gilt es zu unterscheiden zwischen einer Makro- und einer Mikroebene. Die Mikroebene umfasst dabei alle dynamischen Prozesse, die die einzelnen Entwicklungsstufen eines Textes vom leeren Blatt Papier bis zum letzten Bearbeitungsschritt beschreiben. Sie sind also auf der Ebene eines einzelnen Textzeugen anzusiedeln. Dabei sollten alle textuellen Veränderungen, die sich in einzelne Prozesse segmentieren lassen, identifiziert, separat kodiert und nach Möglichkeit in eine chronologische Abfolge gebracht werden. Auf der untersten Ebene sind hier die einzelnen Modifikationsprozesse zu nennen, die sich aus Streichungen, Hinzufügungen oder Umstellungen des Textes ergeben. Hier ist es nicht immer leicht, eine genaue Reihenfolge festzulegen. Ein paar Indizien, die Auskunft darüber geben können, existieren dennoch. So lässt sich z.B. recht gut erkennen, ob ein Autor eine Textstelle sofort beim Schreiben geändert hat oder erst zu einem späteren Zeitpunkt. Eine Sofortkorrektur wird sich in der Regel dadurch identifizieren lassen, dass der ersetzende Text unmittelbar hinter der Streichung notiert wurde. Ein Ersetzen des Wortes zu einem späteren Zeitpunkt wird dagegen normalerweise dadurch gekennzeichnet sein, dass der ersetzende Text nicht unmittelbar in den Zeilenfluss integriert ist, sondern über der gestrichenen Passage oder am Rand notiert wurde.³⁵⁸ Die Prozessabfolge besteht in beiden Fällen anscheinend erst einmal aus denselben Operationen: 1. Einfügen eines Wortes, 2. Streichen und 3. Einfügen der Ersetzung. Dies stimmt aber für das zweite Beispiel nur, wenn dieser Ersetzungsprozess separat betrachtet wird. Der eigentliche Prozessablauf im zweiten Fall kann zusätzlich zwischen der ersten und der zweiten Operation noch eine Reihe weiterer Operationen beinhalten, das Einfügen weiterer Worte (evtl. auch noch weitere Modifikationsprozesse) bis zu dem Punkt, wo besagter Text ersetzt wurde. Je genauer der Schreibprozess sequenziert werden kann, desto genauer ist dieser Prozess auch rekonstruierbar. Dabei könnte der Entstehungsprozess eines Textes beispielsweise auch in Sequenzen auf Wort- oder Buchstabenebene eingeteilt werden. So ließe sich die Textgenese z.B. Buchstabe für Buchstabe verfolgen, quasi wie eine Kamera, die dem Verfasser über die Schulter schaut. Dabei stellt sich natürlich die Frage, welchen Erkenntnisgewinn eine so feingranulare Sequenzierung bringen würde. Dass beim Schreiben eines Wortes Buchstabe auf Buchstabe folgt,

³⁵⁷ LEBRAVE: Hypertexts (wie Anm. 355), S. 223.

³⁵⁸ Natürlich existieren hier auch eine Reihe von Ausnahmen. So wäre etwa vorstellbar, dass Segmente in einem mit Bleistift geschriebenen Text auch nach Beendigung des kompletten Schreibprozesses ausradiert worden sein könnten und sich der in die so entstandenen Lücken eingefügte Text dann doch wieder im Zeilenfluss befände oder ein maschinenschriftlicher Text mit Tippex korrigiert wurde.

führt wohl zu keinem sichtbaren Mehrwert. So erscheint es sinnvoller, den Prozess in größere Abschnitte zu unterteilen, etwa in unterschiedliche Bearbeitungsphasen, die mehrere Prozesse bündeln. Dies hat den Vorteil, dass sich Prozesse, die sich nicht in eine chronologische Reihenfolge bringen lassen, in Phasen eingeteilt werden können, die sich ihrerseits wiederum voneinander abgrenzen lassen. Dies können sowohl Phasen sein, die von einer einzelnen Person stammen, als auch solche, an denen mehrere Personen beteiligt sind. Mögliche Einteilungen können sich z.B. aus der Bündelung von Prozessen einzelner Akteure ergeben, wie dies im Beispiel der Nuntiaturberichte Pacellis der Fall ist. Diese Prozesse können im Idealfall stark sequentiell angeordnet sein. Hier sind die Bearbeitungsstufen der unterschiedlichen Akteure in der Regel festgelegt (Aufnahme des Textes durch einen Stenotypisten, Überarbeitung durch einen Sekretär und abschließende Korrekturen durch Pacelli). Natürlich kann es aber auch hier zu Abweichungen kommen und zu unterschiedlichen Bearbeitungsabfolgen. Es ist nicht immer eindeutig zu unterscheiden, welche Bearbeitungsschicht auf welche folgt, fest steht aber, dass es zeitlich abgrenzbare Schichten sind. Die Anordnung der Schichten ist dann Deutung und nur eine mögliche Rekonstruktion von vielleicht vielen. Dennoch macht es Sinn, diese Prozesse zu segmentieren, da sie auch später noch in eine beliebige Reihenfolge gebracht werden können. Eine andere Unterteilungsmöglichkeit wäre z.B. die nach zeitlich abgrenzbaren Bearbeitungsphasen eines Autors. In Abbildung 2.19 sind solche Prozesse modellhaft skizziert.

Auf der unteren Achse sind in Level und Steps unterteilt die Veränderungsprozesse an einem Textträger dargestellt. Die erste Unterteilung, die Level beschreiben identifizierbare Bearbeitungsphasen (in diesem Beispiel gekennzeichnet durch unterschiedliche Schreiber), die Veränderungen an einem Dokument vorgenommen haben. Alle Veränderungsprozesse innerhalb einer Phase sind durch die Step-Markierungen gekennzeichnet. Die einzelnen Dokumente symbolisieren in diesem Modell keine unterschiedlichen Textzeugen, sondern einen einzelnen, der von einer oder mehreren Personen bearbeitet wurde. Freilich könnte ein konkretes Beispiel auch lediglich aus zwei Level mit zugehörigen Steps bestehen, etwa dann, wenn nur eine Person an dem Entstehungsprozess beteiligt ist, oder mehrere Personen, bei denen eine Einteilung in einzelne Phasen aber nicht als sinnvoll erachtet wird. Andererseits könnte auch die Textgenese eines einzelnen Verfassers in mehrere Level unterteilt werden, z.B. dann, wenn man kennzeichnen möchte, dass es sich um eine frühe, mittlere oder späte Bearbeitungsphase handelt. Auch denkbar wäre es, einzelne Level in Einheiten zusammenzufassen, um weitere aussagekräftige Unterteilungen vornehmen zu können. Auf der Zeitachse werden die Geneseprozesse chronologisch eingeordnet. Auch wenn eine exakte chronologische Gliederung der einzelnen Prozesse nicht immer möglich ist, so stellt ein Hinzufügen eines Geneseschrittes theoretisch zumindest immer einen eigenen Wert t auf der Zeitachse dar. Im Idealfall sollte ein System zur Kodierung und Darstellung von dynamischen Prozessen in der Lage sein, zu jedem beliebigen Zeitpunkt t den zu diesem Zeitpunkt gültigen Zustand eines Textes darstellen zu können.³⁵⁹

Auf der Makroebene hingegen werden genetische Prozesse beschrieben, die sich zwischen unterschiedlichen Textträgern abspielen. Einzelne Textzeugen können innerhalb

³⁵⁹ Gleichsam einer Pausetaste, die den aktuellen Schreibprozess einfriert.

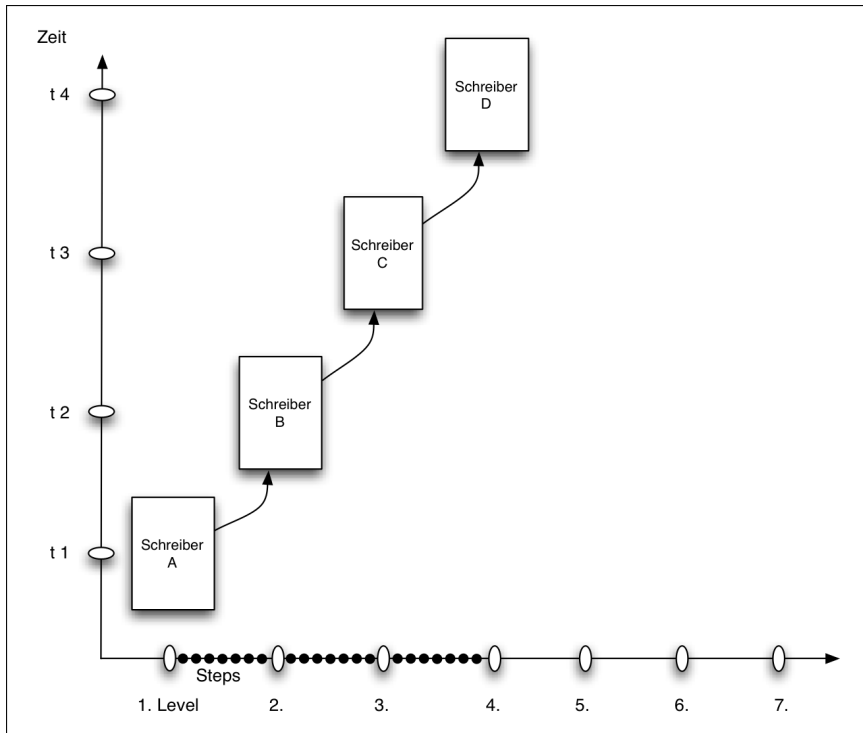


Abbildung 2.19 Binnenvariante Textgenese

einer außenvarianten Textgenese auf einer höheren Stufe wiederum in eine chronologische Abfolge gebracht werden. Die Abfolgen können teilweise sehr einfach, schwierig oder auch gar nicht rekonstruierbar sein. So ist z.B. eine einfache lineare Abfolge von überarbeitetem Entwurf und anschließender Fixierung des Textes in einer Ausfertigung möglich, wie das schon öfter erwähnte Pacelli-Beispiel zeigt. Es handelt sich dabei um unterschiedliche Textzeugen, die aufgrund ihrer Datierung und Produktionspraxis ohne Zweifel chronologisch eingeordnet werden können. Was die bis jetzt beschriebenen Beispiele gemeinsam haben, ist die Tatsache, dass als Ziel in der Regel ein abschließender Text steht. Die Entwürfe dienen hier in der Regel nur als Vorstufe zu einem Text, sie sind also schon bei ihrer Erschaffung darauf angelegt, nicht mehr zu sein als ein Zwischenstadium auf dem Weg zu einem Endstadium. Meistens ist die Textgenese aber nicht so einfach zu rekonstruieren. Vor allem dann nicht, wenn die Entwicklung nicht linear verläuft und wenn viele Textzeugen an der Entstehung beteiligt sind. Am schwierigsten auf der Makroebene ist es sicherlich, die Textgenese von Überlieferungen zu rekonstruieren, die nicht zu dem Zweck geschrieben wurden, nur als Zwischenstufe zu einem finalen Text zu fungieren. Ein gutes Beispiel dafür ist die Edition der *Canterbury Tales*. Die Textgenese solcher Textzeugen beschreibt den Prozess, der in der klassischen Druckedition durch Variantenapparate dargestellt wurde und vor allem in Editionen antiker und mittelalterlicher Überlieferungen zu

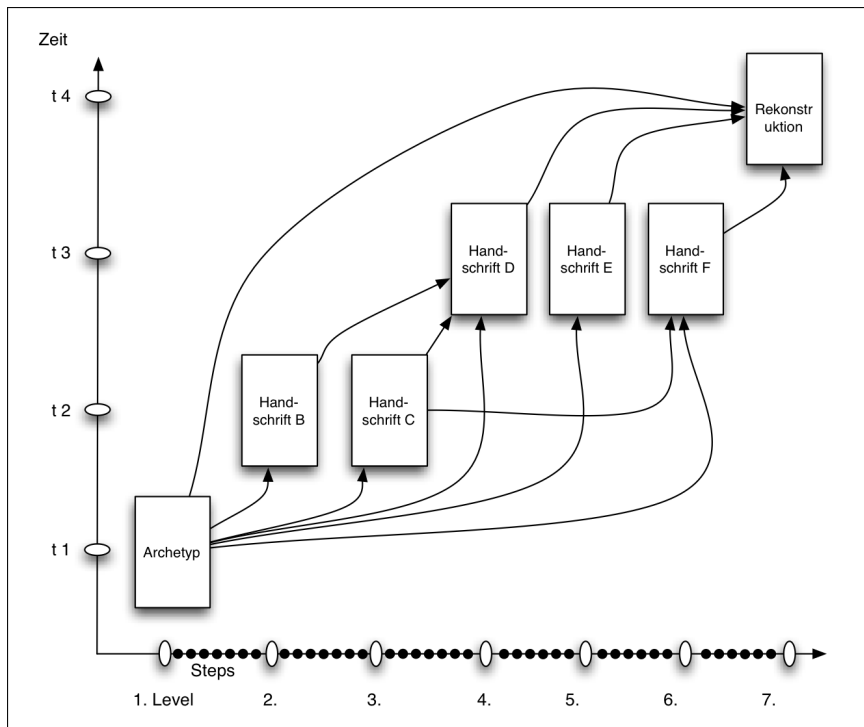


Abbildung 2.20 Binnenvariante Textgenese

finden ist. Früher war es das Ziel, durch verschiedene Ansätze, Kontaminationen eines Textes zu bereinigen und die Textgestalt, die dem Ursprungstext (wenn es diesen denn überhaupt gab) am ähnlichsten war, wiederherzustellen.³⁶⁰ Heutzutage haben sich eher die beschreibenden, nicht rekonstruierenden Konzepte der Verwandtschaft oder der Ähnlichkeit durchgesetzt. In erster Linie wird versucht, die Veränderungsprozesse zu beschreiben, die sich bei der Abschrift von Texten ergeben haben. Dies ist schwierig, da man diese Änderungen dem Text allein nicht ansieht. Man benötigt in der Regel die Variantentexte als Vergleichsobjekte. Hier unterliegt die Segmentierung der textgenetischen Prozesse stark der Interpretation des Editors. Deswegen ist es angebracht, die einzelnen Textzeugen als prinzipiell gleichwertig zu betrachten und nicht einen fiktiven Ursprungstext rekonstruieren zu wollen, sondern die unterschiedlichen Varianten in synoptischer Darstellung zu präsentieren, um dem Benutzer eine eigene Interpretation zu ermöglichen. Die vom Editor vorgeschlagene Lesart kann dabei als eine mögliche Alternative von vielen angesehen werden. In der Theorie bilden diese Textzeugen aber dennoch Einheiten, die sich auf einer Zeitachse darstellen lassen.

Ein stark vereinfachtes Modell dieser Art dynamischer Textprozesse ist in Abbildung 2.20 zu sehen. Dabei können am Ausgangspunkt und am Endpunkt der Text-

³⁶⁰ http://www.uni-saarland.de/fak4/fr41/germanistik/pdfs/studieninfos/infos_neueStO/Info_Basiswissen_LitWiss.pdf S.9

genese durchaus auch mehrere Textzeugen stehen, wenn zum Beispiel der Ursprungstext nicht nur aus einem, sondern aus mehreren Textzeugen besteht oder zeitgleich verschiedene 'Endfassungen' eines Textes entstanden sind. Die einzelnen Textzeugen können auf der Mikroebene natürlich auch wieder aus genetischen Prozessen bestehen. Hier können also, anders als bei der binnenvarianten Textgenese, zu einem beliebigen Zeitpunkt *t* auch mehrere parallele Textzustände existieren.

2.5.2 Chronologische Lesarten

Hier gilt es zu unterscheiden zwischen der tatsächlichen chronologischen Abfolge des eigentlichen Schreibprozesses (Produktionschronologie) und der vom Autor beabsichtigten Leseabfolge (Rezeptionschronologie). Diese Chronologien können, müssen aber

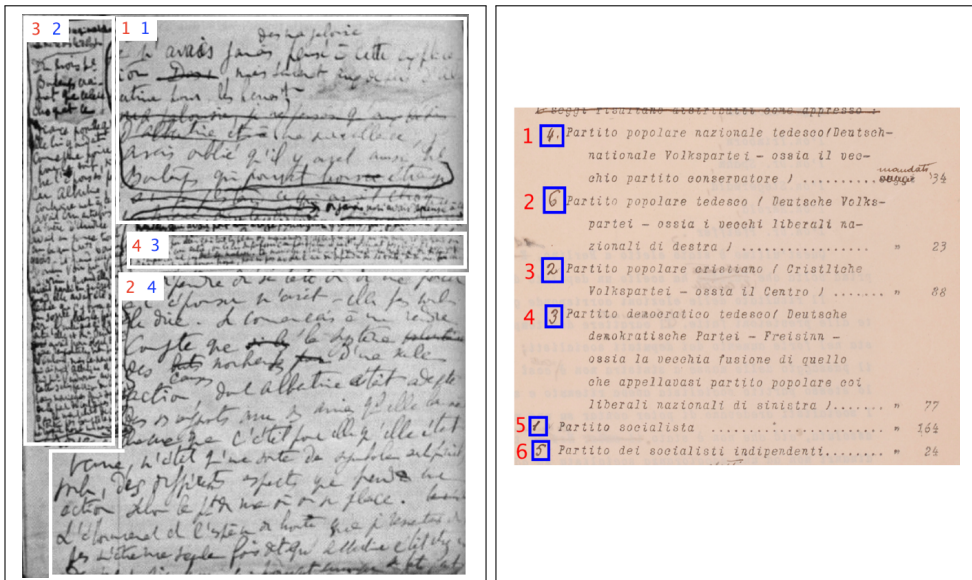


Abbildung 2.21 Rekonstruktion unterschiedlicher Chronologien (rot = Produktionschronologie, blau = Rezeptionschronologie) links: Proust-Manuskript, rechts: Entwurf Eugenio Pacellis

nicht übereinstimmen, wie in Abbildung 2.21 zu sehen.³⁶¹ Um beide sichtbar zu machen, könnte man ein unterschiedliches Zählsystem (Listing 2.2) verwenden.³⁶²

³⁶¹ Im vorliegenden rechten Beispiel handelt es sich um einen maschinenschriftlichen Entwurf zum Wahlergebnis der deutschen Nationalversammlung von 1919. Der Grundtext wurde durch einen unbekannten Stenotypisten maschinenschriftlich erfasst und später durch handschriftliches Einfügen einer neuen Sortierreihenfolge ergänzt. Das Wahlergebnis wird dadurch nach der Anzahl der erreichten Stimmen neu sortiert (Metamarkierungen zu Beginn jeder Zeile). Die nicht mehr nachzuweisende Ausfertigung ist in Analogie zu den bekannten Nuntiaturberichten erneut von einem Stenotypisten maschinenschriftlich verfasst worden. Sie enthält mit großer Sicherheit die Tabelle mit der geänderten Sortierung.

³⁶² Das Listing 2.2 bezieht sich auf das Proust-Beispiel in Abbildung 2.21 links. Manuskriptbeispiel entnommen aus: <http://www.proustian.com/images/manuscript%20page-a.jpg>


```

1 <div levelWriting="1" levelReading="1">...</div>
2 <div levelWriting="2" levelReading="4">...</div>
3 <div levelWriting="3" levelReading="2">...</div>
4 <div levelWriting="4" levelReading="3">...</div>

```

Listing 2.2

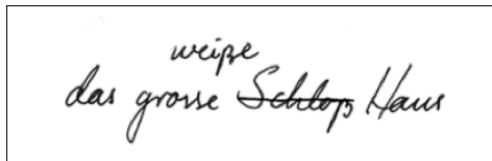


Abbildung 2.22

Die Produktionschronologie und die Rezeptionschronologie müssen dabei nicht zwangsläufig einem vorgeschriebenen Weg folgen, sondern hängen von der jeweiligen Interpretation ab, sie müssen also alternative Entstehungsabfolgen zulassen.³⁶³ So hat die von Huitfeldt beschriebene Problema-

tik der Lesarten in Abbildung 2.22³⁶⁴ eine zusätzliche Dimension, wenn man die verschiedenen möglichen Abfolgen der Modifikationsprozesse betrachtet. Neben den beiden von Huitfeldt identifizierten Lesarten

1. das grosse weisse Haus
2. das grosse weisse Schloß

existieren zusätzlich Lesarten in Bezug auf die Chronologie des Schreibprozesses. In vorliegendem Beispiel ist nicht eindeutig zu klären, ob zuerst 'Schloß' durch 'Haus' ersetzt und dann 'weisse' eingefügt wurde, oder ob zuerst 'weisse' eingefügt und dann 'Schloß' durch 'Haus' ersetzt wurde. Folgende chronologische Varianten sind also im Prinzip möglich:

1ab. das grosse Schloss

2a das grosse Haus

oder

2b das grosse weisse Schloss

3ab. das grosse weisse Haus

Zieht man bei diesem Beispiel zusätzlich die Möglichkeit in Betracht, dass es sich bei 'weisse' auch um eine alternative Lesart zu 'grosse' handeln könnte, führt dies zusätzlich zu den rezeptionschronologischen Varianten:

³⁶³ Vgl. dazu Pichler. Für ihn liegt das Potential Digitaler Editionen vor allem darin, „to make the types of interpretation - and their differences - explicit and extractable, to give the user the possibility to choose between the different Level of interpretation, and to realize them in different ways.“ PICHLER: Transcriptions (wie Anm. 31), S. 692.

³⁶⁴ Beispiel entnommen aus http://www.tei-c.org/About/Archive_new/ETE/Preview/huitfeldt.xml

1c das grosse Schloss
 2c das grosse Haus
 oder
 1d das weiße Schloss
 2d das weiße Haus

Diese unterschiedlichen chronologischen Abfolgen können also auch zu unterschiedlichen Lesarten führen. In einem XML-basierten Markup-System könnte dies folgendermaßen gelöst werden:

```

1  Das
2  <add levelWriting="a0 b0 c0">grosse</add>
3  <add levelWriting="a3 b1 d0">weiße</add>
4  <del levelWriting="a1 b2 c1 d1">Schloss</del>
5  <add levelWriting="a2 b3 c2 d2">Haus</add>.
```

Listing 2.3

Diese Kodierung würde alle Informationen enthalten, um z.B. ein dynamisches stufenweises Einblenden aller chronologischen Varianten zu ermöglichen. Beginnt ein Level mit 0, wird diese Aktion direkt zu Beginn bei der ersten Stufe ausgeführt ('grosse' ist also in allen drei chronologischen Varianten a, b und c von Beginn an sichtbar, nicht aber bei d):

- | | |
|--------------------------|-----------------------------|
| a) | b) |
| 0. Das grosse Schloss | 0. Das grosse Schloss |
| 1. Das grosse | 1. Das grosse weiße Schloss |
| 2. Das grosse Haus | 2. Das grosse weiße |
| 3. Das grosse weiße Haus | 3. Das grosse weiße Haus |
| c) | d) |
| 0. Das grosse Schloss | 0. Das weiße Schloss |
| 1. Das grosse | 1. Das weiße |
| 2. Das grosse Haus | 2. Das weiße Haus |

Ein Zählsystem muss aber noch über weitergehende Möglichkeiten verfügen. Es muss untergliederbar sein (siehe Abschnitt 2.5.1), um verschiedene Modifikationsabfolgen innerhalb unterschiedlicher Geneseschichten durchnummerieren zu können. Das in dieser Arbeit verwendete technische Textmodell soll mehr als die herkömmlichen Modelle den prozesshaften Charakter beschreiben. Deshalb soll in diesem Modell auch der Schwerpunkt auf sequenzierbaren Einzelprozessen liegen. Die wesentliche Einteilung beruht auf den zwei hauptsächlichen dynamischen Operationen des Einfügens und des Entfernens von Elementen innerhalb eines Dokumentes. In diesem Sinne sind

auch Umstellungen oder Veränderungen von Sortierreihenfolgen auf ihre einzelnen Modifikationsschritte heruntergebrochen als kombinierte Prozesse von Entfernen und Hinzufügen zu verstehen.

Das in dieser Arbeit verwendete Textverständnis ist ein prozedurales. Eine zentrale Annahme geht davon aus, dass sich dynamische Prozesse (zumindest theoretisch) immer durch eine chronologische Abfolge von Modifikationen beschreiben lassen. So besitzen Dokumente zu jedem Zeitpunkt ihres Bestehens einen definierten Zustand, der sich von einem Zustand zu einem Zeitpunkt davor und/oder danach unterscheidet. Jede Operation muss daher mit einer Art Zeitstempel versehen und in Einzelprozesse segmentiert werden können.³⁶⁵ Da eine absolute Chronologie nur in den seltensten Fällen rekonstruierbar ist, erscheint ein Zählsystem, das die relative Aufeinanderfolge von Prozessen kodiert, sinnvoll. Die Chronologisierung der Prozesse wird durch ein hierarchisches System von Einheiten und Untereinheiten beschrieben, so dass sich zeitliche Modifikationsgruppen bilden lassen. Um eine ausreichende Erschließungstiefe zu gewährleisten, müssen die Gruppierungen beliebig tief segmentierbar sein.³⁶⁶ Ein solches Levelsystem funktioniert dabei ähnlich wie die Untergliederung von Kapiteln und Unterkapiteln:

1. Schicht 1
 - 1.1
 - 1.2
2. Schicht 2
 - 2.1
 - 2.2 Unterschicht 2.2
 - 2.2.1
 - 2.2.2
 - 2.2.3
 - alternative Chronologie/Lesart zu Unterschicht 2.2
 - 2.2.3
 - 2.2.2
 - 2.2.1
 - 2.3
3. Schicht 3
 -

Zusätzlich muss ein Mechanismus bestehen, der alternative Abfolgen, wie die der unterschiedlichen Produktions- und Rezeptionschronologie kodierbar macht (im vorigen Listing die alternative Chronologie/Lesart zu Unterschicht 2.2, die hier die Abfolge aus Unterschicht 2.2 umkehrt, also eine chronologische Lesart verzeichnet). Im praktischen Teil der Arbeit wird ein solches Gliederungssystem verwendet, um die

³⁶⁵ Zu einem ähnlichen Schluss gelangt auch schon Pierazzo. Elena PIERAZZO: The Encoding of Time in Manuscript Transcription: Toward Genetic Digital Editions, in: Digital Humanities 2007, Conference Abstracts 2007, S. 150–152, URL: <http://www.ideals.uiuc.edu/handle/2142/2368>.

³⁶⁶ Gruppierungen von Änderungsprozessen, die nicht auf chronologischen Kategorien basieren, lassen sich z.B. durch die Typisierung oder Klassifizierung von Modifikationsprozessen bilden. Hier sollte das System flexibel erweiterbar sein, um eigene Gruppierungskategorien bilden zu können.

Modifikationsprozesse sortieren zu können. Gleichzeitig wird auf dieser Grundlage ein Navigationsmechanismus zur Verfügung gestellt, der alternative Lesarten eines Textes ermöglicht.

2.6 Synoptische Ansichten

Die Annahme Lachmann, dass es sich bei Varianten zumeist um 'korrupte' Textzeugen handelt, die zwangsläufig in einer korrigierten Version münden sollten, hat sich gewandelt. Gerade Digitale Editionen vermögen es, diesen Fakt zu hinterfragen und die prinzipielle Gleichwertigkeit von Varianten und Textstufen zu postulieren. Nicht zuletzt die visuellen Möglichkeiten, Textstufen und Varianten nebeneinander zu präsentieren, haben zu diesem Umdenken geführt. Viele Digitalen Editionen warten mit der Möglichkeit auf, Transkriptionen, digitale Faksimile, diplomatische Ansichten, Lesetexte etc. in parallelisierter Darstellung nebeneinander darzustellen. Zumeist werden ganze Textzeugen oder Abschnitte parallelisiert. Ein bekanntes Werkzeug, welches die parallel-segmentation method³⁶⁷ der TEI als Grundlage nutzt, ist die Versioning-Machine (zur Zeit in der Version 4.0).³⁶⁸ Visualisiert werden die unterschiedlichen Transkriptionen horizontal nebeneinander (2.23). Um z.B. alle varianten Textstellen innerhalb einer Zeile vergleichen zu können, müssen erst alle Textzeugen abschnittsweise geöffnet werden, um dann mühselig jede Zeile des Textzeugen vergleichen zu können. Bei einer Textüberlieferung mit zahlreichen varianten Fassungen stößt ein solches Vorgehen schnell an seine Grenzen.

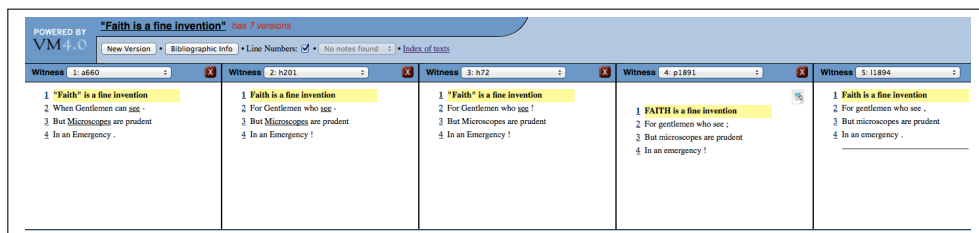


Abbildung 2.23 Beispielsynopse der Versioning-Machine v4.0

Woran es hier fehlt, ist eine zeilengenaue dynamische Einblendung der Varianten zu dem Zeitpunkt, an dem man diese angezeigt bekommen möchte. Dies wird versucht zu umgehen, indem beim Anklicken einer Zeile die entsprechende Textstelle des varianten Textzeugen mit einem Highlighting versehen wird, was das schnellere Auffinden aber nur unwesentlich beschleunigt. Darüber hinaus wird lediglich auf die Richtlinien der TEI verwiesen, wie die Daten zu kodieren sind, um diese Ansicht zu generieren. Ein System, dass es ermöglicht, Varianten annotieren zu können, ohne von dem Benutzer zu verlangen, Markup-Strukturen in Form von XML schreiben zu müssen, existiert nicht.

³⁶⁷ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPPS>

³⁶⁸ <http://v-machine.org/samples.php>

Insgesamt wird bei der Darstellung Digitaler Editionen ein zu großes Gewicht auf die sequentielle Verarbeitung von XML-Daten. In der Regel werden die gesamten Dokumentstrukturen serverseitig mit Hilfe von XSL-Stylesheets generiert. Eine Modifikation der Ansicht, wie z.B. das parallele Einblenden von Varianten, wird dann mit dem gleichen serverseitigen Prozess ausgeführt und die komplette Ansicht erneut aufgebaut, statt nur die entsprechenden Bildausschnitte oder Textfragmente anzupassen. Gerade bei diesen Prozessen werden die Möglichkeiten des Document-Object-Models und dynamische Programmiertechniken wie Ajax zu selten genutzt. Der Nutzer hat meist zu wenige Einflußmöglichkeiten auf das Aussehen und die Anordnung der Informationen. Möchte er nur die vorhandenen Varianten zu einem bestimmten Textausschnitt konsultieren, wird er es im Allgemeinen als lästig empfinden, erst alle varianten Textzeugen öffnen zu müssen und das Display mit unnötigen Informationen zu belasten. Ähnlich skeptisch äußerte sich auch schon Bradley in Bezug auf die sequentielle Verarbeitung von XML durch SAX-Parser (Simple API for XML).³⁶⁹ Das Problem des ereignisbasierten Modells³⁷⁰ bestünde darin, dass es XML-Daten sequentiell verarbeite, ohne die Möglichkeit zu berücksichtigen, dass während des Prozesses auf andere Elemente zugegriffen werden müsse. Änderungen eines XML-Elements können sich oft auch auf Elemente an anderen Stellen der Baumstruktur auswirken. Diese Kritik äußert er u.a. auch an Systemen wie Tustep. Das Vorgehen dort sei ähnlich. Er nennt dies 'sweep-through/fix-up' - also zuerst einen Durchlauf über die gesamte Datenmenge zu initiieren und in einem zweiten Schritt Korrekturen an den einzelnen Datenstrukturen vorzunehmen und den Prozess so lange zu wiederholen, bis alle Modifikationen abgearbeitet sind. Bei diesem Vorgehen fehle es an der Einbeziehung von kontextbezogener Informationen, die schon zur Laufzeit berücksichtigt werden sollte. Ein System, das diese nicht-ereignisbasierten Mechanismen unterstützt, muss also mehr als ein einfacher XSLT-Prozessor in der Lage sein, direkt auf andere Elemente zugreifen zu können. Hier unterstreicht er die Möglichkeiten, die das DOM-Interface in diesem Bezug bietet, da es den gesamten Dokumentbaum im Speicher vorhält und Methoden zur Verfügung stellt, um einzelne Teile des Baumes dynamisch zu manipulieren. Darüber hinaus ist das Document Object Model auch besser geeignet, um auf Inhalte 'fremder', nicht XML-basierter Applikationen, zuzugreifen und deren Inhalte in die Baumstruktur zu integrieren.³⁷¹

Die Informationskodierung sollte auch nicht in Abhängigkeit von einem zukünftigen Publikationsformat oder aufgrund schon bestehender Editionspraktiken erfolgen, da dies keine abstrahierte Form der Datenverwaltung ermöglicht und zu vielen parallelen Verzeichnungspraktiken führt. An dem Beispiel des apparatus-Moduls der TEI wird dies besonders deutlich. Es existieren drei unterschiedliche Grundtechniken, Varian-

³⁶⁹ John BRADLEY: Tools to augment scholarly activity: an architecture to support text analysis, in: *Augmenting Comprehension: Digital Tools and the History of Ideas 2004*, hrsg. v. D. BUZZETTI/G. PANCALDI/H. SHORT, S. 19–48, 19ff.

³⁷⁰ Margit BECHER: XML – DTD, XML-Schema, XPath, XQuery, XSLT, XSL-FO, SAX, DOM, 2009, S. 294.

³⁷¹ „The Document Object Model is a platform- and language-neutral interface that will allow programs and scripts to dynamically access and update the content, structure and style of documents.“ <http://www.w3.org/DOM/>

ten mit Hilfe dieses Moduls zu implementieren: location-referenced method, double end-point attached method, parallel-segmentation method.³⁷² Diese können wiederum (teilweise) als location external oder internal realisiert werden. Es besteht also keine genormte Vorgehensweise im Sinne eines eindeutigen Standards. Die TEI betont auch ausdrücklich, dass sie hier keine Regeln vorgeben möchte.³⁷³ Dieses Verhalten führt allerdings zu dem Problem, dass TEI-verarbeitende Softwaresysteme auf jede Form der so erzeugten XML-Strukturen eine 'passende Antwort' parat haben müssen, also die Kodierungslogik ebenso implementieren müssen. Zudem sind die Methoden nicht vollständig kompatibel zueinander. Die double end-point attachment method kann überlappende Varianten abbilden,³⁷⁴ wohingegen dies mit der parallel-segmentation method nicht möglich ist.³⁷⁵ Somit ist eine Übersetzung eines Apparates in einen anderen ohne Informationsverlust nur in ganz bestimmten Situationen möglich.³⁷⁶

Die TEI scheint insgesamt noch stark von der Druckkultur inspiriert zu sein und passt sich zu sehr den dort gebräuchlichen Editionsmethoden an. Es ist wahrscheinlich eine Konzession an die vielen schon gedruckt vorliegenden Variantenapparate und deren Kodierungstechniken, wie die Autoren selbst im Abschnitt zur location-referenced method angeben: „The location-referenced method of encoding apparatus provides a convenient method for encoding printed apparatus; in this method as in most printed editions, the apparatus is linked to the base text by indicating explicitly only the block of text on which there is a variant.“³⁷⁷ Diese Art der Kodierung bietet aber keine eindeutigen Positionsangaben auf die Textstelle (Lemma), zu der der Apparateintrag die Variante markiert: „When the apparatus is linked to the text by means of location references, as shown here, it is not possible to find automatically the precise portion of text varied by the readings.“³⁷⁸ Dies wird durch die zusätzliche Notierung eines lemma-Elements zu lösen versucht (Listing 2.4). Das lem-Element mit dem Inhalt 'Experience' soll dabei eine Referenz auf das Wort Experience in <ln="1» ermöglichen. Wie ungenau diese Methode allerdings ist, wird deutlich, wenn man z.B. annimmt, dass in der selben Zeile noch einmal das Wort Experience ver-

³⁷² <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPLK>

³⁷³ „The identification of discrete textual variations or apparatus entries is not a purely mechanical process; different editors may group readings differently. No rules are given here as to how to group readings into apparatus entries; the tags given here may be used to group readings in whatever way the editor finds most perspicuous or useful.“ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPEN>

³⁷⁴ „This method is designed to cope with 'overlapping lemmata':“ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPDE>

³⁷⁵ „In this method, no two variations can overlap.“ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPPS>

³⁷⁶ „Where double-end-point-attachment encodings have no overlapping lemmata, translation of these to the parallel segmentation encoding and back will also be possible without loss of information“, ebd. Wobei das in diesem besonderen Fall nicht der TEI anzulasten ist, was ausdrücklich betont werden soll, sondern den Umständen, dass die parallel-segmentation method in erster Linie für die Kodierung von Varianten verwendet werden sollte, in der kein Textzeuge oder keine Rekonstruktion als Leittext vorgesehen sind. Dafür müssen alle varianten Textstellen synchronisiert werden (also dürfen sich nicht überlappen), um eine Austauschbarkeit untereinander zu gewährleisten.

³⁷⁷ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPLO>

³⁷⁸ Ebd.

zeichnet ist. Man müsste dafür das entsprechende Textsegment in `<l n="1">` wiederum mit einem Tag versehen und über einen Pointer mit `<lem>` verbinden, was wiederum einen sehr hohen Kodierungsaufwand bedeuten würde. Insofern ist diese Methode (ohne Referenz-Mechanismus) inkompatibel zur double end-point attachment method.

```

1 <l n="1">Experience though noon Auctoritee
2 <app>
3   <lem wit="#E1">Experience</lem>
4   <rdg wit="#La">Experiment</rdg>
5   <rdg wit="#Ra2">Eryment</rdg>
6 </app>
7 </l>
8 <l>Were in this world ...</l>

```

Listing 2.4

Die Komplexität und der Aufwand all dieser Kodierungstechniken werden selbst von der TEI nicht in Frage gestellt und es wird des öfteren davon abgeraten, die Texte manuell auszuzeichnen.³⁷⁹ Ein System, das alle unterschiedlichen Techniken unterstützt, existiert zur Zeit allerdings nicht. Überhaupt scheint hier die Einstellung der TEI etwas fragwürdig. Man zieht sich gerne auf die Position zurück, die TEI biete genügend Mechanismen an, um die gewünschte Information zu kodieren, der Rest wäre Sache der Applikation. Diese Einstellung verschiebt die Problematik aber nur auf die Seite der Anwendungsprogrammierung. Es ist nicht einfach, Applikationen für einen Standard zu implementieren, der kaum den Anforderungen eines Standards entspricht. Für das Apparat-Modul würde das bedeuten, dass eine Applikation alle verschiedenen Kodierungstechniken beherrschen muss und dabei noch sicher zu stellen hat, dass diese nur in dem jeweiligen (wie auch immer gearteten) Bestimmungskontext angewendet werden dürfen.

Diese Nachteile zeigen, dass eine abstraktere Form der Kodierung anzustreben ist. Die Art und Weise, wie etwas dargestellt wurde oder werden soll, darf nicht zur Grundlage des Datenmodells werden. Es muss eine eindeutige, neutrale Kodierung erfolgen aus welcher die entsprechenden unterschiedlichen Outputs generiert werden können, ob dies nun eine Liste der Textzeugen zu Beginn eines Buches oder auf einer separaten Seite der Digitalen Edition ist oder die Positionierung der Varianten direkt innerhalb des Textes oder separat davon mit der location-referenced method.

³⁷⁹ „Because creation and interpretation of double end-point attachment apparatus will be lengthy and difficult it is likely that they will usually be created and examined by scholars only with mechanical assistance.“ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPDE>

2.7 Retrieval

Beim Retrieval innerhalb textgenetischer Editionen kann es zu unterschiedlichen Problemen kommen, da es sich bei diesen Texten nicht um fixierte Momentaufnahmen eines Textes zu einem bestimmten Zeitpunkt handelt, sondern um eine Vielzahl von Textschichten, die meist zu unterschiedlichen Zeitpunkten entstanden sind. Im Prinzip stellt jede Modifikation eines Textes auf einer Zeitachse t aus Sicht einer Suchanfrage einen neuen Text dar (vgl. Abschnitt 2.5.1). Aus Sicht XML-basierter Kodierungen kann es dabei zu folgenden Problemen kommen: Eine Datenstruktur wie der Satz:

```
1 <l>das große <del type="striketrough">weiße</del> Haus</l>
```

Listing 2.5

muss in einer Phrasensuche sowohl ein Ergebnis für 'das große weiße Haus' als auch für 'das weiße Haus' liefern, da die erste gültige Textinstanz die Streichung nicht beinhaltet, während die zweite, die sich auf der Zeitachse nach der ersten befindet, sie enthält bzw. das gestrichene Wort nicht enthält. Dies ist durch multiple Indizes auch durchaus realisierbar. In gängigen XML-Datenbanken wie eXist kann ein Index sowohl über das komplette line-Element erstellt werden, inklusive der Inhalte aller Kindelemente (hier über `<l>`), als auch über den Inhalt unter Exklusion bestimmter Elemente (``). Somit wären mit zwei unterschiedlichen Indizes beide Suchvarianten möglich. Diese Methode funktioniert allerdings nicht mehr, wenn die Strukturen komplexer werden, wie Listing 2.6 zeigt:

```
1 <l>das große
2   <del rend="striketrough">weiße</del>
3   <add place="above"><del type="striketrough">blaue</del></add>
4   <add type="instantCorr" place="above">grüne</add>
5   Haus
6 </l>
```

Listing 2.6

Diese Struktur würde folgende textgenetische Sequenz kodieren:

1. das große weiße Haus
2. das große blaue Haus
3. das große grüne Haus

Alle drei Textstrings müssten exakt so auffindbar sein. Bei einer Indexerzeugung unter Exklusion der `del`- oder `add`-Elemente würde nicht mehr das erwünschte Ergebnis

erzielt werden können, weil damit alle drei Sequenzen ausgeschlossen wären und eine Suche nur ein Ergebnis für 'das große Haus' liefern würde. Bei Inklusion aller del- und add-Elemente wäre nur die komplette Sequenz indexierbar: 'das große weiße blaue grüne Haus'. Eine Exklusion nur der add-Elemente würde zu dem Ergebnis 'das große weiße Haus' führen und eine Exklusion der del-Elemente nur zu 'das große grüne Haus'. Auch wenn man alle diese unterschiedlichen Indizes erstellen würde, so würde doch der in der Mitte liegende Fall 'das große blaue Haus' durch keinen Index abgedeckt werden. Dies unterstreicht einmal mehr die Notwendigkeit, die einzelnen Prozesse mit einem Zählsystem zu versehen, welches alle Sequenzen rekonstruierbar und adressierbar macht. Dies könnte folgendermaßen aussehen:

```

1 <l>das große
2   <del rend="strikethrough" version="1">weiße</del>
3   <add place="above" version="2"><del>blaue</del></add>
4   <add type="instantCorr" place="above" version="3">grüne</add>
5   Haus
6 </l>

```

Listing 2.7

Die textgenetischen Sequenzen wären hier nun anhand der Level indexierbar. Zuerst würde ein Index über den gesamten Text unter Exklusion aller Elemente, die version != 1 enthalten zu dem Ergebnis 'das große weiße Haus' führen, ein zweiter unter Exklusion aller Elemente version != 2 zu 'das große blaue Haus' und ein dritter mit der Exklusion version != 3 zu 'das große grüne Haus'. Im Gegensatz zu dem in Abschnitt 2.5.2 vorgestellten Zählsystem zur Nummerierung der Modifikationsprozesse handelt es sich hier lediglich um eine Hilfsnummerierung für einen Indexierungsprozess.

Problematisch wird es auch dann, wenn Techniken der Kodierung eingesetzt werden, um multiple Hierarchien (Überlappungen) zu realisieren, wie etwa bei Fragmentierungen oder Milestones. XML-basierte Datenbanken indexieren in der Regel die Inhalte von Elementen oder Attributen. Bei Milestones sind die Elemente leer, bei Fragmentierungen werden sie an 'unnatürlichen' Grenzen unterbrochen. Wird ein Textstring etwa an folgenden Grenzen fragmentiert:

```

1 <l><frag1>das <frag2>große</frag2></frag1> <frag2>weiße Haus</frag2></l>

```

Listing 2.8

und möchte man aber trotzdem innerhalb von frag2 nach 'große weiße Haus' suchen (da es ja weiterhin eine semantische Einheit bildet und nur aus Gründen der Überlappung fragmentiert werden musste), führt dies in der Regel zu einem leeren Resultat. Dies mag bei Fragmentierungen, die formattechnischer Natur sind ein kleineres

Problem sein (rte-Editoren von Browsern etwa arbeiten bei überlappenden Formatierungen nach diesem Prinzip). Deren semantischer Gehalt ist meist gering und man durchsucht in der Regel nicht solche Inhalte. Bei einer Fragmentierung eines Elements, welches semantisch gesehen eine Einheit bildet, ist diese Suchmöglichkeit aber unbedingt erforderlich. Es ist zwar denkbar, den String mit einer Volltextsuche über das parent-Element <l> zu ermitteln, aber das schließt die Möglichkeit aus, gezielt den Inhalt nur eines spezifischen Elementtyps auszuwerten. Es besteht nun zwar die Alternative, diese Elemente über Techniken wie xPointer oder xLink zu adressieren (von milestone:start bis milestone:end etwa oder fragment:next), allerdings sind die meisten XML-Datenbanken nicht in der Lage, Inhalte über diese Techniken zu indizieren.³⁸⁰ Erstreckt sich ein Textstring also über zwei Fragmentierungsgrenzen, ist dieser ohne Modifikationen der Index-Engine oder des Tokenizers der jeweiligen Datenbank nicht auffindbar, bzw. müssen selbst Tokenizer implementiert werden, die den entsprechenden Index erzeugen.

Weitere Suchoptionen, die einen Mehrwert bedeuten könnten, wären Abfragen bezüglich der Positionierung von Textsegmenten auf Textzeugen, z.B. finde alle Suchterme XYZ, die sich als Randnotizen des Schreibers A oberhalb des Haupttextes (wenn ein solcher definiert werden kann) befinden. Dafür reicht eine ungenaue Typisierung der Textelemente in Attributform, wie etwa „marginal note“ nicht aus. Hier wäre es sinnvoll, Suchmechanismen zu implementieren, die Textsegmente nach absoluten (auf das gesamte digitale Faksimile bezogen) oder relativen (bezogen auf andere Textsegmente oder Objekte – Images beispielsweise) Koordinaten referenzieren können.

2.8 Multiple Hierarchien

Wie schon mehrfach angedeutet, spielen multiple Hierarchien eine zentrale Rolle in textgenetischen Editionen. Da die Kodierung topographischer Informationen eine große Bedeutung bei der genetischen Analyse spielt und logische Strukturen in der Regel diesen zuwiderlaufen,³⁸¹ stellt sich die Frage nach der adäquaten Kodierung. Das klassische Linebreak-Beispiel, bei dem sich Zeilenumbrüche nicht mit logischen Strukturen vereinbaren lassen, ist hier nur eines von vielen, da ja nicht nur der Zeilenumbruch, sondern auch die unterschiedlich angeordneten Schreibräume und die Modifikationen der Streichungen, Hinzufügungen und Umstellungen von Zeichenketten alle in der diplomatischen Grundtranskription verzeichnet werden sollten. Hier wird es zwangsläufig zu konkurrierenden Hierarchien kommen, die sich nicht 'gleichwertig' mit Hilfe eines hierarchischen Markup-Systems beschreiben lassen.³⁸² Hinzu kommen die unter-

³⁸⁰ Die Arbeit bietet XML-basiert auch keine Lösung für dieses Problem, es sollte aber bei der Verwendung von hierarchischem Markup nach dem OHCO-Modell immer im Auge behalten werden.

³⁸¹ „Die topografische Struktur liegt stets quer zu der abstrakten 'logischen' Struktur des Textes aus Linearität des Zeichenbestandes und Hierarchie der Inhaltsobjekte – ist aber gerade aus editorischer Sicht ein wesentlicher Schlüssel zur Konstitution des Textes“ SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 174.

³⁸² Vgl. hierzu auch Brüning et al., die sich aufgrund dieser Einschränkungen in ihrem Editionsprojekt zu Goethes Faust zu einer radikalen Trennung der Transkriptionsebenen entschlossen

schiedlichen möglichen chronologischen Lesarten, wie in Abschnitt 2.5.2 beschrieben. Auch hier kann es wiederum zu Überschneidungen der semantischen Ebenen kommen. Unterschiedliche Interpretationen solcher Lesarten führen nicht selten zu überlappenden Strukturen (Listing 2.9).

```

1 <produktionschronologie schreibraum="A">
2   Text 1.1
3   Text 1.2
4 <rezeptionschronologie xml:id="1" target="#2"/>
5   Text 1.3
6 </produktionschronologie>
7 <produktionschronologie schreibraum="B">
8   Text 2.1
9   Text 2.2
10 <rezeptionschronologie xml:id="2"/>
11   Text 2.3
12 </produktionschronologie>

```

Listing 2.9

Auch auf der Makrostrukturebene sind bei komplexen Überlieferungen immer wieder solche Phänomene zu beobachten, wie das Beispiel in Abbildung 2.24 zeigt.³⁸³ Insgesamt stellt sich also die Frage, ob hierarchisches Markup prinzipiell das geeignete Werkzeug ist, in textgenetischen Editionen diese Strukturen zu beschreiben.³⁸⁴ Ein alternatives Konzept³⁸⁵ besteht darin, stand-off-basierte Kodierungen zu verwenden, um die multiplen Hierarchien modellieren zu können.³⁸⁶ Die TEI bietet über die XInclude-Technik einen solchen Mechanismus an. Damit ist es z.B. möglich, beliebige Knoten innerhalb eines XML-Trees mit xpointern zu adressieren und somit für ei-

haben: „In particular, the encoding was impaired by an increasing number of conflicts, many of them caused by overlapping hierarchies [...]. When it finally proved unavoidable, we decided to split the encoding into two distinct transcripts: a documentary transcript of each page of a manuscript and a textual transcript of the whole manuscript.“ Das führt allerdings wiederum zu der Problematik, die unterschiedlichen Transkriptionen konsistent zu halten: „Yet there is identical content in these two TEI documents and with it the threat of inconsistency“ BRÜNING/HENZEL/PRAVIDA: *The Case of 'Faust'* (wie Anm. 327), §18, §32. Je mehr Ebenen auf diese Art und Weise voneinander getrennt werden, desto schwieriger wird es allerdings, die Konsistenz zu gewährleisten.

³⁸³ Das Beispiel stammt aus dem Kapitel 12.2.2 'The double end-point attachment method' der TEI-Richtlinien (<http://www.tei-c.org/release/doc/tei-p5-doc/en/html/TC.html#TCAPDE>). Die hier dargestellte Visualisierung der Überlappungen über Linewidgets wird im Abschnitt 3.2.2 ausführlich beschrieben.

³⁸⁴ Claus HUITFELDT: *Multi-Dimensional Texts in a One-Dimensional Medium*, in: *Computers and the Humanities* (28 4.5) 1995, S. 235–241, hier S. 239.

³⁸⁵ Neben den in Abschnitt 1.2.2 erwähnten Möglichkeiten.

³⁸⁶ James CUMMINGS: *Converting Saint Paul: A new TEI P5 edition of The Conversion of Saint Paul using stand-off methodology*, in: *Literary and Linguistic Computing* (24.3) 2009, S. 307–317, Vgl.

ne Basistranskription unterschiedliche Markup-Strukturen zu erzeugen.³⁸⁷ Allerdings besteht der Nachteil dieser knotenbasierten Methode, die mit der Funktion `range()` innerhalb des `xpointer`-Statements ausgeführt wird, darin, dass Referenzknoten für die Einfügepositionen im Zieldokument vorhanden sein müssen, und das der Output des `XInclude`-Statements eine valide XML-Struktur sein muss, was hier die Behandlung von Überlappungen erschwert. Dies bedeutet, dass schon bei dem Design des Hauptdokumentes darauf geachtet werden muss, die kleinstmögliche Basis einer überlappenden Hierarchie („markup for the smallest significant units of analysis“³⁸⁸) mit XML-Elementen auszuzeichnen, um diese adressieren zu können. So vorausschauend wird wohl keine Transkription kodiert werden können, da im Prinzip an jeder einzelnen Zeichenposition eine Informationseinheit beginnen könnte. Eine weitere Methode besteht in der Möglichkeit, über `string-range()` die Zeichenpositionen (relativ zu einem Elternelement) direkt zu adressieren. Hier ist allerdings zu beachten, dass diese Methode nur auf Zeichendaten und nicht auf Knoten anwendbar ist, d. h., dass alle XML-Elementknoten innerhalb des ranges nicht ins Zieldokument übernommen werden.

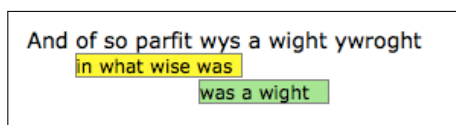


Abbildung 2.24

Die Vorteile stand-off-basierter Methoden insgesamt bestehen darin, dass keine Informationsschicht eine bevorzugte Stellung einnimmt, der Nachteil, dass sich Hierarchien nicht aus der Markup-Struktur selbst ergeben, sondern durch andere zusätzliche Methoden implementiert werden müssen. Insofern verhält sich

stand-off-Markup ähnlich wie die Kodierung von XML-Daten mit start- und end-Milestones. Auch hier werden hierarchiefreie Ankerpunkte gesetzt, die den Textrange angeben, auf den sich die Informationen der Milestones beziehen. Dennoch scheint die stand-off-Methode angesichts der Möglichkeiten, die Kodierung vom eigentlichen Text trennen zu können, in vielen Szenarien sehr brauchbar zu sein.³⁸⁹ Mit ihr lassen sich Daten auszeichnen, die aus verschiedenen Gründen keine zusätzlichen Markup-Strukturen beinhalten dürfen. Dies kann z.B. sinnvoll sein, wenn eine im WWW verfügbare Transkription, die innerhalb eines Editionsprojekts entstanden ist, mit interpretativem Markup versehen werden soll, aber keine Veränderungen an der originalen Transkription erlaubt sind.³⁹⁰ Auch ist eine stärkere Modularisierung³⁹¹ der Daten möglich, die eine saubere Trennung der Schichten und somit einen effizienteren Zugriff zum Zweck der Weiterverarbeitung und Analyse erlaubt: „This way, the text, kept in as neutral form as possible, remains an attractive resource, open to future analyses and

³⁸⁷ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SA.html#SASOso>

³⁸⁸ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SA.html#SASOso>

³⁸⁹ Vgl. Desmond SCHMIDT: The Role of Markup in the Digital Humanities, in: Historical Social Research / Historische Sozialforschung (37.3) 2012, S. 125–146.

³⁹⁰ Vgl. CUMMINGS: Saint Paul (wie Anm. 386).

³⁹¹ „Descriptions of properties belonging to different theoretical perspectives are expected to be separate, in order to constitute separate modules that can be judged, verified and challenged on their own“ BANSKI: Why (wie Anm. 72).

to the creation of new views.“³⁹² Die unterschiedlichen Sichten (views) ergeben sich nicht zuletzt aus unterschiedlichen Interpretationen. In diesem Sinne ist eine Modularisierung durchaus erstrebenswert. Die Kodierung aller Informationsschichten, sind sie nun innerhalb eines Editionsprojekts verifiziert worden oder bestehen sie aus zusätzlichen Schichten, die im Laufe späterer Analysephasen hinzukommen, innerhalb eines embedded Markup-Systems zu verzeichnen, würde zu immer komplexeren Dokumenten führen, die schwieriger zu bearbeiten, zu verwalten und zu analysieren wären, als modular getrennt kodiert vorliegende Schichten. Die Menge der theoretisch vorhandenen Schichten und die Limitierung von hierarchischem Markup, diese konkurrierenden Schichten zu modellieren, lässt ernsthafte Zweifel daran entstehen, dass dies mit den Methoden von OHCO-basierten Markup-Systemen noch adäquat zu leisten wäre.³⁹³ Ein Vorteil dieser Modelle soll ja gerade darin bestehen, offen für Fragestellungen zukünftiger Generationen von Forschern zu sein, und es wird auch immer wieder betont, dass einmal ausgezeichnete Informationsschichten in anderen Forschungsprojekten als Grundlage weiterer Analysen dienen können.³⁹⁴ Hier wäre es allerdings viel sinnvoller, nur jeweils die Schichten aus einer stand-off-Struktur zu serialisieren, die auch von Interesse für die jeweilige Fragestellung sind. Die Trennung von textuellen Daten (als Basisdokument) und Markup-Strukturen (in stand-off-Dokumenten) kann auch sehr sinnvoll sein, wenn nur die Inhalte der Markup-Strukturen von Interesse sind, wenn also nur die informationsbeschreibenden Schichten zu Analysezwecken benötigt werden und ihr content zweitrangig ist. Aufgrund der unzureichenden Modellierungsmöglichkeiten in Bezug auf überlappende Strukturen und der modularen Vorteile wurde in dieser Arbeit ein eigenes stand-off-System verwendet, das auf Zeichenebene die Positionen innerhalb von Zeilenelementen kodiert (mehr dazu siehe Abschnitt 3.2.12).

2.9 Usability

Die Usability Digitaler Editionen kann auf unterschiedlichen Ebenen betrachtet werden. Zum einen auf der Seite der Kodierung. Dies bedeutet oftmals einen Spagat zwischen größtmöglicher Offenheit (also einer neutralen Kodierung, die unterschiedliche Fragestellungen an die Edition ermöglicht³⁹⁵) und einer Kodierung, die durch ein

³⁹² Ebd.

³⁹³ Vgl. dazu auch die Markup-Experimente Fiormontes zu den textuellen Varianten Valerio Magrellis: „Many (but not all) of the phenomena described here can be represented only with great effort or with an unacceptable level of imprecision. XML, like its predecessor SGML, was originally designed as an instrument for archiving and information retrieval, whereas the mouvance of the text poses challenges for the encoding and representation that can only be resolved by the development of a different model of encoding, and by the design of an adequate user interface on the application level“ Valentina Schmidt Desmond FIORMONTE Domenico Martiradonna: Digital Encoding as a Hermeneutic and Semiotic Act: The Case of Valerio Magrelli, in: Digital humanities quarterly (4.1) 2010, URL: <http://www.digitalhumanities.org/dhq/vol/4/1/000082/000082.html>, §35.

³⁹⁴ Vgl. CUMMINGS: Saint Paul (wie Anm. 386).

³⁹⁵ Nach dem Prinzip, eine Edition „ könne allen möglichen Zwecken gleichzeitig dienen.“ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 133.

bestimmtes Erkenntnisinteresse geleitet ist und eine genaue Analyse der Benutzererwartung voraussetzt, da eine bestimmte Nutzergruppe eben ganz spezifische Fragestellungen an die Edition hat oder um es mit den Worten von Czmiel zu formulieren: „The crucial point in the process of designing the data structure should be that different scholars have different intellectual requirements from resources.“³⁹⁶ Zum anderen kann die Usability aber auch ganz praktisch auf der Ebene des Benutzers betrachtet werden: Wie ist die Bedienbarkeit der Edition? Wie gelange ich zu den Informationen, die ich benötige? Welche Navigationselemente stehen mir zur Verfügung? Welche unterschiedlichen Ansichten auf die Editionstexte und Digitalisate kann ich mir anzeigen lassen? Inwieweit kann ich diese selbst verändern?

Die Praxis der TEI, dass bei der Kodierung entschieden werden kann, für welchen Ausgabezweck ein Apparat angewendet werden soll, ist kritisch zu betrachten und das Gegenteil von dem, was heutzutage auch unter dem Stichwort *user-driven* bezeichnet wird.³⁹⁷ Eine solche Methode berücksichtigt nämlich nicht, dass in einem anderen 'Bestimmungskontext' eine andere Kodierung wünschenswerter wäre.³⁹⁸ Wären alle Apparattechniken der TEI ineinander automatisch transformierbar, wäre die Kodierung der unterschiedlichen Techniken obsolet und man könnte eine Grundtechnik zum Standard erheben. Die Generierung der anderen Apparatmodelle wäre dann nur noch eine Frage der jeweiligen Applikation und könnte dynamisch aus dieser Grundkodierung gewonnen werden.³⁹⁹ Allgemein stellt sich auch die Frage nach der Handhabbarkeit von XML- und TEI-basierten Kodierungspraktiken. Bei digitalen Editionsprojekten muss immer berücksichtigt werden, wie technikaffin die Mitarbeiter des Editionsprojekts sind und ob der Aufwand, der mit dem Annotieren nach den Kodierungs-Richtlinien verbunden ist, auch vertretbar ist. Auch heutzutage gibt es einen nicht geringen Prozentsatz von Editionswissenschaftlern, die hier mit Berührungsgängsten zu kämpfen haben. So gibt Romary etwa zu bedenken: „There remains a portion of such scholars who intuitively consider the TEI not entirely appropriate

³⁹⁶ Alexander CZMIEL: *Editio ex machina - Digital Scholarly Editions out of the Box*, in: *Digital Humanities* 08 2008, S. 101–102, URL: www.ecl.oulu.fi/dh2008/Digital%20Humanities%202008%20Book%20of%20Abstracts.pdf, hier S. 101.

³⁹⁷ „It is argued that there is not so much a gap between print and electronic medium but much more a transition in a thinking process: from the output orientation of the classical style of editing towards a data- and user-driven new approach.“ Malte REHBEIN: *The transition from classical to digital thinking. Reflections on Tim McLoughlin, James Barry and collaborative work*, in: *Jahrbuch für Computerphilologie* 10 2008, URL: computerphilologie.digital-humanities.de/jg08/rehbein.pdf, S. 1.

³⁹⁸ Hier liegt also Rehbein nicht ganz richtig, wenn er behauptet: „The TEI provides guidelines for text encoding, but not necessarily for editing principles.“ ebd., S. 7, weil eben genau diese Kodierung der TEI versucht, die editorischen Praktiken der Variantenverzeichnung abzubilden. (Vgl. Abschnitt 2.6)

³⁹⁹ Bradley sieht hier z.B. die Möglichkeit benutzerunabhängig mit DOM-Objekten zu arbeiten und auf Anfrage den gewünschten output zu erzeugen. Mit dem DOM wäre es möglich, den output dynamisch aus einer Grundtranskription so zu generieren, dass er den Anforderungen des users entspricht und nicht durch den Entwurf des Dokumentes vorgegeben ist. Das hätte den Vorteil, dass wiederum andere Applikationen auf diesen temporären output analytisch zugreifen könnten: „The control of what the user is allowed to do with the text rests entirely with the server owner, who generally provides for the kinds of queries that represent the lowest common denominator of access.“ BRADLEY: *Tools* (wie Anm. 369), S. 35.

for them, sometimes even fearing that adopting the TEI may result in more more trouble than benefit to their research project.“⁴⁰⁰

Eine zu starke Fixierung auf bestimmte Kodierungsstandards führt häufig nicht zu den gewünschten Ergebnissen bzw. schöpft nicht das volle Potential der Online-Editionen aus, weil diese Formate unterschiedlichen Einschränkungen unterliegen oder keine Lösung für spezifische Szenarien anbieten.⁴⁰¹ Dennoch kann es sinnvoll sein, sich an 'Standards', wie sie beispielsweise die TEI bietet, zu orientieren, da es als Austauschformat zwischen Applikationen fungieren kann. Das ist aber zu großen Teilen davon abhängig, wie gut sich die Daten mit diesem Format beschreiben lassen. Sollte der Informationsverlust dabei zu groß sein oder zu viele Kompromisse eingegangen werden müssen (wie etwa bei einer unzureichenden Kodierung der Varianten mit der location-referenced method – Abschnitt 2.6), ist davon abzuraten. Es gibt auch andere gute Gründe, auf standardisierte Formate zu verzichten, z.B. bei der Speicherung in DBMS. Genauso, wie es bei relationalen Datenbanken gute Gründe geben kann, nicht alle Normalisierungsregeln auf das Datenmodell anzuwenden, um z.B. eine höhere Verarbeitungsgeschwindigkeit zu erreichen, kann es innerhalb von XML-Datenbanken sinnvoll sein, datenbankspezifischen Code zu generieren, um die XML-Struktur für bestimmte Funktionalitäten zu optimieren. Auf die Verarbeitungsgeschwindigkeit oder die Indexierung von XML-Content kann z.B. schon die Grundsatzentscheidung, Informationseinheiten in Attributen oder Elementen zu kodieren, entscheidenden Einfluss haben. Bis dato ist dem Autor keine Datenbank bekannt, die den umgekehrten Weg beschreitet und ihre Datenverarbeitung für speziell entwickelte Formate, wie z.B. das der TEI, optimiert. Insgesamt steht also die Frage im Raum, wer sich wem anpassen sollte? Das verwendete Markup (Datenmodell) den spezifischen Anforderungen eines wie immer gearteten Standard? Oder ist es nicht sinnvoller, applikationsoptimierte Datenstrukturen in ihrem jeweiligen Anwendungskontext zu verwenden und die Austauschbarkeit über ein anderes standardisiertes Austauschformat, wie z.B. das der TEI, zu gewährleisten. Allerdings müsste die TEI sich dann mehr in die Richtung eines wirklichen Standards bewegen und nicht versuchen, alle möglichen (disziplinenabhängigen) Auszeichnungstechniken als gültigen TEI-Code zuzulassen. Dann wäre es auch für Anwendungsprogrammierer einfacher, die XML-Daten in ein jeweiliges applikationsspezifisches Format zu transformieren.

Die zweite Ebene der Usability betrifft, wie eingangs schon erwähnt, die Benutzbarkeit der Edition selbst und im wesentlichen auch die des Systems, mit welchem die Edition erstellt wird, da die in dieser Arbeit implementierte browserbasierte Applikation ja auch als kollaboratives System einsetzbar sein soll, das den Benutzer auch gleichzeitig zum Produzenten macht. Dies umfasst für das Gebiet der textgenetischen Digitalen Editionen vor allem folgende Punkte:

⁴⁰⁰ Laurent ROMARY: Questions and Answers for TEI Newcomers, in: Jahrbuch für Computerphilologie 10 2008, URL: <http://computerphilologie.digital-humanities.de/jg08/romary.pdf>, S. 1.

⁴⁰¹ Keating z.B. sieht dies ähnlich und beschreibt seine Entscheidung für ein selbstdefiniertes XML-Format, anstatt die Richtlinien der TEI zu verwenden, ausführlich. G. KEATING u. a.: Account Book (wie Anm. 87).

- Bildbearbeitungs- und Darstellungsfunktionen.
- Intuitive Kodierungsmöglichkeit der Informationsschichten.
- Intuitive Darstellung von Informationsschichten der Textzeugen und digitalen Faksimile.
- Verknüpfungsmöglichkeiten aller Informationsebenen der Textschichten mit den digitalen Faksimile.
- Diplomatische Transkriptionen mit genauer Texttopographie des Originals.
- Überblendungsmöglichkeiten des Faksimile mit Text.
- Synoptische Darstellungen in unterschiedlichen Varianten (Synopsisen von Text und Faksimile /Text und Text /Faksimile und Faksimile). Diese wiederum mit den einzelnen Möglichkeiten der Zeilen-, Abschnitts-, Seitensynopse.
- Navigationsmechanismen um z.B. chronologische Abfolgen/Lesarten sichtbar zu machen.
- Zitierbarkeit jeder einzelnen Geneseschicht und des Bildausschnittes des digitalen Faksimile.

Teilweise werden einzelne Punkte dieser Liste in unterschiedlichen Anwendungen im Umfeld Digitaler Editionen bereits berücksichtigt. Bildbearbeitungssoftware oder die Parallele Darstellung von Textzeugen etwa sind keine Neuerungen in dem Sinne und werden schon seit langer Zeit bei der Bearbeitung und Darstellung von Editionen eingesetzt. Dennoch existiert bis heute kein System, das all diese Punkte unter besonderer Berücksichtigung textgenetischer Editionen in einem kollaborativen, browserbasierten Werkzeug verbindet. Im folgenden praktischen Teil soll ein solches System vorgestellt werden.

3 System Design

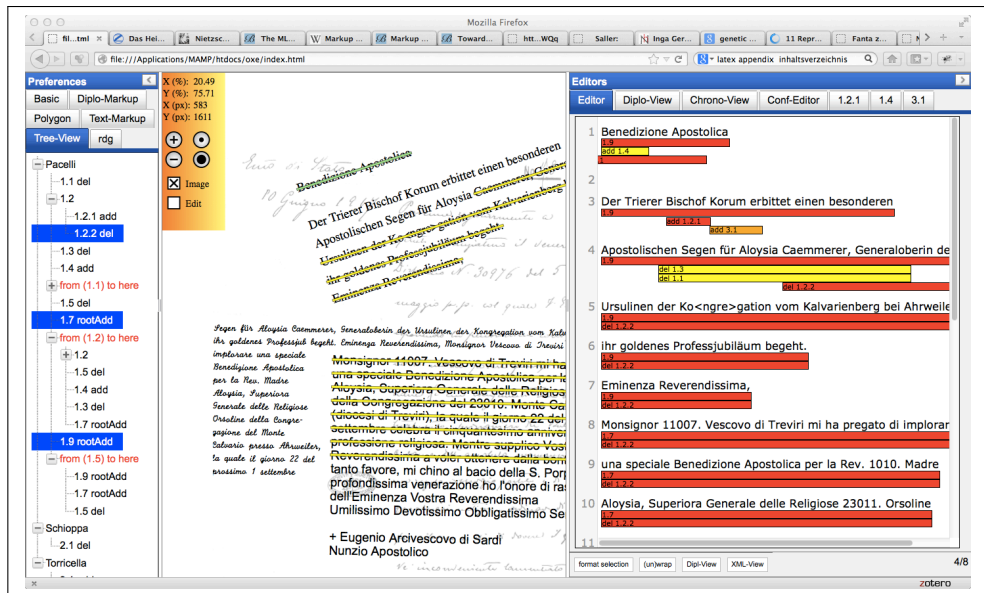


Abbildung 3.1

3.1 Hauptkomponenten

Abbildung 3.1 zeigt die 3 Hauptkomponenten der Applikation. Auf der linken Seite ist ein schmales Panel zu sehen, das unterschiedliche Darstellungs- und Bearbeitungsmenüs enthält. Das Basic-Menü ist dafür vorgesehen, bestehende Projekte zu laden und neue anzulegen (Abbildung 3.2). Die serverseitige Projektverwaltung ist nicht Bestandteil dieser Arbeit und kann nach Belieben implementiert werden. Die Applikation soll nicht an ein bestimmtes DBMS oder Daten-Format gebunden sein. Das in der Applikation verwendete Datenmodell ist DOM-basiert und kann durch geeignete Schnittstellen in das jeweilige gewünschte Format serialisiert werden. Ein Beispiel, wie dies für eine TEI-Serialisierungsschnittstelle aussehen kann, wird in Abschnitt 3.2.13 beschrieben. Stellvertretend für das Laden von Projekten aus einem serverseitigen DBMS werden in diesem Menü verschiedene Beispielprojekte bereit gestellt, die über die entsprechenden Buttons aus der Datei `examples.js` geladen werden können.

Zusätzlich enthält das Menü die Möglichkeit, Images in jpeg- und png-Format von der lokalen Festplatte zu importieren. Diese können dann mit den entsprechenden Werkzeugen annotiert sowie mit den in der Editor-Komponente angelegten Transkriptionen verknüpft und in einem serverseitigen DBMS gespeichert werden.

Ein leeres Projekt entspricht der Grundeinstellung der Applikation beim ersten Aufruf der Seite (kein Image, leeres Editorfenster). Zusätzlich enthält das Menü die Filterfunktionen für die geladenen Images (Näheres dazu in Abschnitt 3.3.1). Die übrigen Menüs beinhalten die Funktionen, die benötigt werden, um die Images zu annotieren (Diplo-Markup zum Erstellen diplomatischer Ansichtslayer und Textüberblendungsfunktionen und Polygon, um Bildausschnitte mit den kodierten Textelementen des Editors zu verbinden), Text-Markup zum Annotieren der Textdaten mit genetischen Informationen, den Navigationsmechanismus Tree-View, um die Genese der Textzeugen zu visualisieren und das rdg-Menü, das eine Auswahlliste der unterschiedlichen Textzeugen enthält, um diese dynamisch ein- und ausblenden zu können. Die einzelnen Menü-Funktionen werden in den folgenden Kapiteln ausführlich beschrieben. In der Mitte befindet sich die sogenannte SVG-View, in die Images geladen und bearbeitet werden. Rechts davon ist die Editor-Komponente zu sehen, bestehend aus dem Haupteditor zur Erstellung der genetischen Transkriptionen und einem Config-Editor um globale oder projektbezogene Einstellungen zu verwalten. Die GUI-Grundarchitektur der Applikation basiert auf dem Layout-Manager des javascript-Frameworks YUI.⁴⁰² Die Datenstrukturen in dieser Applikation sind in erster Linie DOM-basiert und somit über die üblichen javascript-Funktionen manipulierbar. Für alle SVG-Objekte (Polygone, um Textausschnitte zu markieren und Textobjekte, die der Erstellung diplomatischer Ansichten und Textüberblendungsfunktionen dienen) bedeutet dies, dass sie vollständig in SVG vorliegen. Die Transkriptionstexte selbst und die Konfigurationsdateien werden in Editorinstanzen verwaltet, die auf Grundlage der Browser-Editor-API Codemirror⁴⁰³ implementiert wurden. Das Text-Markup-Modul ist eine Erweiterung zu diesem Editor und wurde mit Hilfe von Linewidget-Objekten realisiert, die in die Editorinstanzen unter die jeweilige Zeile eingefügt werden. Das Zusammenspiel der einzelnen Komponenten erfolgt vollständig

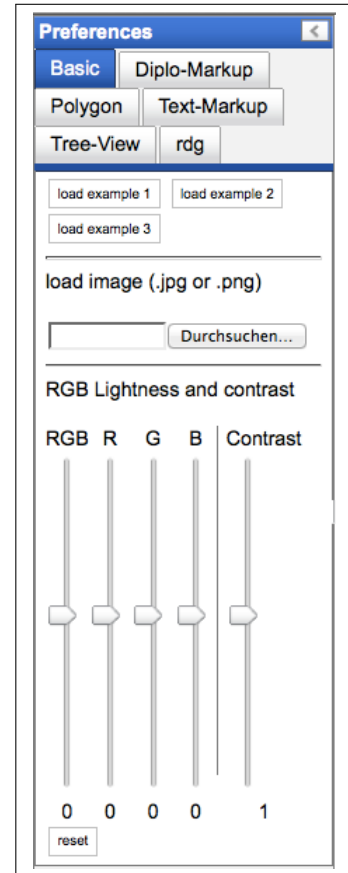


Abbildung 3.2 Projektverwaltung

⁴⁰² <http://developer.yahoo.com/yui/2/>

⁴⁰³ <http://codemirror.net/>

über javascript-Methoden, die auf die einzelnen Komponenten über das DOM zugreifen. Im Folgenden werden diese Komponenten und ihre Funktionsweise beschrieben.

3.2 Editor

Die Art der Speicherung der Informationsstrukturen und die Darstellung dieser ähneln dem linguistischen Annotationssystem Catma (Computer Aided Textual Markup and Analysis). Catma verfolgt hier abseits des üblichen XML-basierten Inline-Markup-Ansatzes eine stand-off-Variante, die Textdokumente mit unterschiedlichen Markup-Strukturen versehen kann.⁴⁰⁴ Die Vorteile einer Verwaltung der Daten in stand-off-Manier liegen auf der Hand. Texte, die mit komplexem Markup versehen werden müssen, wie das bei der Kodierung textgenetischer Prozesse oft der Fall ist, enthalten viele miteinander konkurrierende Hierarchien, die sich häufig überlappen können. Deswegen wurde in diesem System Wert darauf gelegt, multiple überlappende Hierarchien zu unterstützen und möglichst einfache Mechanismen zur Verfügung zu stellen, die dem Bearbeiter keine schwierig zu kodierenden Milestone- oder Fragmentierungstechniken zumuten. Die in der Forschungsliteratur beschriebenen Nachteile stand-off-kodierender Systeme⁴⁰⁵ fallen in diesem Zusammenhang weniger ins Gewicht, da die stand-off-Verwaltung nur das Datenmodell der Applikation betrifft und durch die Möglichkeit der Serialisierung in inlinebasierte Kodierungsmodelle transformiert werden kann. An dieser Stelle soll aber ausdrücklich betont werden, dass der Autor dieser Arbeit auch zahlreiche Vorteile in der Verwendung solcher Modelle als 'dauerhafte' Speicherlösung sieht. Die Trennung von Text und semantischer Information auf diese Art und Weise kann besonders bei der Verwaltung umfangreicher Informationsschichten von Vorteil sein. Hierbei erweist sich nämlich die Stärke inlinebasierter Methoden, die Information direkt an der Stelle des Textes zu kodieren, die sie betrifft, nicht selten als Nachteil. Die oftmals angeführte Behauptung, XML-Daten sind menschenlesbar, trifft hier nur prinzipiell zu. In der Realität sind komplexe Texte, die mit inlinebasierten XML-Strukturen ausgezeichnet wurden in der Regel nur äußerst schwer les- und bearbeitbar.⁴⁰⁶ Auch kann eine solche Trennung zu effizien-

⁴⁰⁴ <http://www.catma.de/>. In erster Linie dient dieses Tool zum Auszeichnen von Texten zum Zweck der linguistischen Analyse. Texte lassen sich mit selbst definierbaren Tagsets annotieren und daraufhin analysieren. Es verfügt nicht über die Möglichkeit Texte zu bearbeiten. Einmal mit Markup versehene Texte können nicht mehr verändert werden, da ihr Markup über das stand-off-System fix an die Zeichenpositionen der jeweiligen annotierten Textstelle gebunden ist.

⁴⁰⁵ Vgl. hierzu auch Sahle: „Vielleicht liegt die einfache Lösung aber auch hier in einem Mittelweg, der zwischen Text-Repräsentierendem markup auf der einen Seite und interpretativem äußeren markup auf der anderen Seite unterscheidet. Das letztere muss natürlich nicht in den Text eingebettet sein (wie die Metadaten eines Header-Bereiches es ja auch nie sind!). Zu beachten sind dann allerdings mögliche Probleme, die sich aus der Veränderung des 'Textes' und der Verschiebung, Differenzierung und Löschung der Referenzpunkte für das nicht eingebettete markup im Text ergeben können.“ SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 250.

⁴⁰⁶ Dies erweitert die von Dahlström beschriebene Problematik des 'Drowning by Versions' auch um einen ganz praktischen Aspekt der Bearbeitung von Texten. Mats DAHLSTRÖM: Drowning by Versions, in: Human IT. Tidskrift för studier av IT ur ett humanvetenskapligt perspektiv 2000, URL: <http://etjanst.hb.se/bhs/ith/4-00/md.htm>.

terer Speicherverwaltung führen. Texte können so z.B. dynamisch mit den nur jeweils gewünschten Informationsschichten versehen und die einzelnen Markup-Dateien so auch auf verteilten Systemen verwaltet werden. Es soll aber auch nicht unerwähnt bleiben, dass dies wiederum das Problem mit sich bringt, die Informationsschichten bei Veränderungen des Basistextes zu synchronisieren, da alle Zeichenpositionen der stand-off-Dokumente (zumindest alle, die der veränderten Textstelle folgen) aktualisiert werden müssen.⁴⁰⁷ Allerdings ist hier auch zu überlegen, ob eine Synchronisierung aller Schichten bei einer Veränderung überhaupt erwünscht ist. Wird ein Text etwa von verschiedenen Bearbeitern oder maschinellen Prozessen mit unterschiedlichen Informationsschichten versehen und wird dieser Text dann von einer anderen Person oder einem maschinellen Prozess verändert, kann dies auch Einfluss auf die übrigen Informationsschichten haben. Hier ist vielmehr zu fragen, ob ein Text nach dem Kodieren seiner Informationsschichten und daran anschließenden Modifikationen überhaupt noch synchronisierbar mit diesen ist? Das hängt natürlich stark von der Art der Veränderung des Textes ab. Eine einfache orthographische Korrektur wird keinen großen Einfluss auf Schichten haben, die nicht primär zum Zweck einer Analyse solcher Phänomene angelegt wurden. Sofern sie aber die Semantik von anderen Schichten betreffen, kann der Einfluss recht groß sein. Wird z.B. eine textkritische Beschreibung von Überlieferungsvarianten in einer Schicht kodiert und es finden sich neue variante Textzeugen, die zu einer einer veränderten Basistranskription führen, so müssen im Prinzip auch die textkritischen Analysen der früheren Auszeichnungsschichten geprüft und gegebenenfalls aktualisiert werden. Dies lässt sich wohl in den wenigsten Fällen automatisiert bewerkstelligen. Es gilt also geeignete Instrumente zu entwickeln, die im Sinne von Versionierungssystemen, die einzelnen Schichten an Versionen binden. Bei der Aktualisierung des Basistextes müssten Kriterien gefunden werden, anhand derer sich verifizieren lässt, ob eine vorige Informationsschicht in die neue Version übertragbar ist, oder ob sie erst einer Bearbeitung bedarf.

3.2.1 Grundkonzept des Markups

Die Diskussion um die Usability von Markup-Strukturen bei der Bearbeitung von textuellen Informationen wird vor allem von zwei gegenläufigen Konzepten bestimmt, zum einen von dem klassischen wysiwyg-Prinzip und zum anderen von dem Konzept, das das Kodieren mit Hilfe unformatierter 'roher' Markup-Strukturen bevorzugt. Beide Methoden haben ihre Vor- und Nachteile. Die Verfechter des wysiwyg-Prinzips betonen dabei die Vorteile, die sich daraus ergeben, dass man schon während der Aufnahme der Daten eine ungefähre Vorstellung davon erlangt, wie der Text in der fertigen Publikation einmal aussehen wird. Dies ist aber auch zugleich ein erheblicher Nachteil, weil somit immer nur eine mögliche Darstellungsform berücksichtigt wird, in der Regel die eines typographisch gestalteten linearen Textes, die stark an Aus-

⁴⁰⁷ Vgl. hierzu auch Ore: „this means that stand-off systems either have to rely on advanced systems for synchronizing data and data counts or the data must be frozen or at least under control of the organisation responsible for the stand-off codes.“ Espen S. ORE: Document Markup – Why? How?, in: *Historical Social Research / Historische Sozialforschung* (37.3) 2012, S. 106–124, hier S. 122.

drucksformen der klassischen Druckkultur erinnert. Da im Digitalen Medium aber diese nur eine mögliche Sicht auf den Editionstext darstellt, scheint die Verwendung des wysiwyg-Prinzip lediglich in solchen Situationen sinnvoll zu sein, wo primär eine Ausdrucksform angestrebt wird. So sind z.B. die Ansichten von normalisiertem Lese-text und diplomatischer Abschrift in der Regel sehr unterschiedlich. In dieser Hinsicht bietet das zweite Konzept der ‘rohen’ Markup-Strukturen eine neutralere Sicht auf die Daten. Hier leidet aber vor allem die Lesbarkeit und die Orientierungsfähigkeit stark, wenn komplexere Auszeichnungstrukturen verwendet werden müssen. Unterschiedliche textgenetische Schichten (von unterschiedlichen Bearbeitern etwa) lassen sich so visuell kaum voneinander trennen. In letzter Konsequenz ist hier wohl eine größere Trennung des Markups und der Textdarstellungen zu bevorzugen. Es scheint sinnvoller, die unterschiedlichen Dimensionen eines Textes so zu visualisieren, dass sie schon bei der Eingabe der Daten einen möglichst übersichtlichen Blick auf alle Informationsschichten erlauben. Dabei soll das hier verwendete Konzept einen Mittelweg beider soeben vorgestellten Methoden darstellen. Es soll mehr als die Markup-Variante visuelle Orientierungspunkte liefern, aber gleichzeitig nicht zugunsten einer Ausgabeform die anderen Sichten vernachlässigen. Deswegen wird in dieser Arbeit auch ein Konzept verfolgt, dass die Markup-Strukturen nicht innerhalb des Textflusses verzeichnet, sondern unterhalb des selbigen. Als Orientierungshilfe dienen dabei drei unterschiedliche Komponenten:

1. Genaue Positionierung des Markups unterhalb des Textabschnitts.
2. Verwendung unterschiedlicher konfigurierbarer Farbattribute, um verschiedene Markup-Elemente oder Gruppen von diesen visuell voneinander trennen zu können.
3. Konfigurierbare textuelle Angaben innerhalb der Markup-Widgets, die eine zusätzliche Orientierungshilfe bieten oder Textvarianten beinhalten.

3.2.2 Linewidthgets

Das Grundkonzept des hier verwendeten Codemirror-Editors ist ein zeilenbasiertes. Die Zeilennummerierung befindet sich am linken Rand des Editors. Bei jedem Zeilenumbruch wird eine neue Zeile inklusive Nummerierung eingefügt. Beim Selektieren eines Textabschnitts kann auf ein Rangeobjekt zugegriffen werden, das über die Angaben `lineStart`, `lineEnd`, `charStart`, und `charEnd` Zugriff auf den selektierten Textausschnitt ermöglicht. Zusätzlich verfügt die Programm-API über die Möglichkeit, die exakten Koordinaten von Zeichen relativ zur linken oberen Ecke der eingefügten Editorinstanz zu ermitteln. Dies macht sich die Linewidthgetimplementierung der Anwendung zu Nutzen, um unterhalb jeder Zeile die Markup-Elemente in Form von css-formatierbaren `div`-Elementen einzufügen (Abbildung 3.3).

Das Textmarkup-Modul wird mit HTML-basierten Linewidthget-Elementen visualisiert, die mit Hilfe von verknüpften jQuery-Data-Objekten in einer Art stand-off-Markupstruktur alle Informationen speichern, die beim Inline-Markup in den Elementnamen und Attributen verwaltet werden. Diese Art der Anzeige von Textmarkup

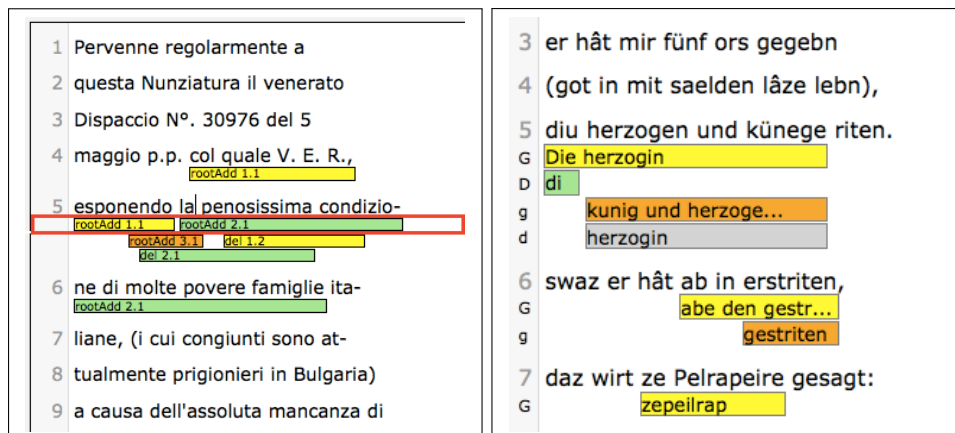


Abbildung 3.3 Unterschiedliche Verwendung von Linewidgets

wurde gewählt, da die oft sehr komplexen Markup-Strukturen textgenetischer Auszeichnungen nach einem Inline-Verfahren zu schwierig zu lesenden und zu bearbeitenden Textansichten führen. Die Linewidgetelemente sind beliebig formatierbar und können mit zusätzlichen Textelementen versehen werden. Sie lassen sich in ihrer Ansicht auf unterschiedliche Art und Weise konfigurieren und ein- und ausblenden.

Linewidgets sind nach dem Prinzip aus Listing 3.1 aufgebaut (Der Ausschnitt zeigt das in Abbildung 3.3 links rot markierte Linewidget). Als Container-Objekte zur Speicherung der einzelnen Markup-Widgets einer jeden Zeile fungieren die Elemente `<div class=„CodeMirror-linewidget“>`,⁴⁰⁸ die alle einzelnen Widgetelemente einer Zeile beinhalten. Auf dieser Hierarchieebene bilden sie parallel zueinander verlaufende Unter-einheiten (sozusagen auch wieder ein Zeilensystem unterhalb einer jeden Textzeile). Dieser gegliederte Aufbau ist deswegen sinnvoll, weil sich somit Markup-Elemente, die sich ganz oder teilweise über den gleichen Textstring erstrecken, nicht überschneiden, sondern auf horizontale Ebenen verteilt untereinander angezeigt werden können. Innerhalb einer jeden so erzeugten Ebene werden die sich nicht überschneidenden Markup-Widgets nebeneinander verteilt.

Die Positionierung der einzelnen Markup-Widgets innerhalb der Linewidgets verfolgt dabei zwei unterschiedliche Konzepte. Das erste ist eine Konzession an die Lesbarkeit des Textes bei textgenetischer Kodierung von binnenvarianten Textphänomenen innerhalb eines Textzeugen (Einfügungen, Streichungen etc.). Dabei wird beim Einfügen eines neuen Widgets nacheinander geprüft, ob eines der Linewidgets unterhalb der Zeile eine freie Einfügeposition zur Verfügung hat und wenn ja, diese genutzt. Ob eine freie Position existiert, wird mit Hilfe eines Linewidget-Positionierungsobjektes ermittelt, das die Positionen neu eingefügter Widgets verwaltet. Existiert keine, wird ein neues Linewidgetelement angelegt und das Markup-Widget dort eingefügt. Somit wird der horizontale Raum effizient ausgenutzt und der

⁴⁰⁸ Alles innerhalb dieser Elemente ist nicht Bestandteil der Codemirror-API und wurde speziell für diese Anwendung implementiert.

Text nicht vertikal in die Länge gezogen. In der Abbildung 3.3 wurde z.B. das zuletzt eingefügte Element del 1.2 der zweiten horizontalen Ebene zwischen rootAdd 2.1 und del 2.1 platziert. Wäre das Widget nicht dort, sondern unterhalb von del 2.1 in einem neuen Linewidget eingefügt worden, hätte dies den Zeilenabstand des eigentlichen Transkriptionstextes auf unnötige Weise vergrößert.

Bei Varianten, die aus unterschiedlichen Textzeugen in paralleler Darstellung angezeigt werden sollen, ist eine andere Art der Verteilung sinnvoller (Abbildung 3.3 rechts). Wie in Zeile 5 zu sehen, werden hier in den Linewidgets zum einen nicht die Elementnamen angezeigt, sondern die varianten Textzeugen und zum anderen erhält jeder Textzeuge ein separates Linewidgetobjekt. Dies dient der Orientierung und der Funktion des zeilenweisen Ein- und Ausblendens aller Varianten eines Textzeugen. Würde auch hier der zur Verfügung stehende Raum genutzt, könnte unterhalb der Zeilennummerierung nicht die übersichtliche Anzeige der jeweiligen Handschrift angezeigt werden können. Auch das zeilenweise Ein- und Ausblenden lässt sich so sehr einfach über die fixe Verbindung der Handschrift mit dem jeweiligen Linewidgetobjekt implementieren (das erste Linewidgetobjekt unterhalb einer Zeile ist z.B. immer für die Handschrift G reserviert).

```

1  <pre>esponendo la penosissima condizio-</pre>
2  <div class="CodeMirror-linewidget">
3      <div id="lineWidget_root_4_0" style="height:11px">
4          <div id="line_4_1" class="widget" onclick="activateWidget(this)"
              onmouseout="resetHighlightWidget(this)" onmouseover="
              highlightWidget(this)" spanid="3_1" style="line-height: 1em; width:
              76.1334px; position: absolute; left: 4px; background-color: rgb
              (255, 255, 0); height: 9px; cursor: pointer; font-size: 9px; border
              : 1px solid black;" color="rgb(255, 255, 0)">rootAdd 1.1</div>
5          <div id="line_4_2" class="widget" onclick="activateWidget(this)"
              onmouseout="resetHighlightWidget(this)" onmouseover="
              highlightWidget(this)" spanid="4_2" style="line-height: 1em; width:
              171.4px; position: absolute; left: 85.05px; background-color: rgb
              (144, 238, 144); height: 9px; cursor: pointer; font-size: 9px;
              border: 1px solid black;" color="rgb(144, 238, 144)">rootAdd 2.1</
              div>
6      </div>
7  <div class="CodeMirror-linewidget">
8      <div id="lineWidget_root_4_1" style="height:11px">
9          ...
10     </div>
11     ...
12 </div>
13 ...
14 </div>

```

Listing 3.1

Die Positionierung der Markup-Widgets wird mit den relativen Koordinaten der Zeichenpositionen der Editorinstanz ermittelt und im jeweiligen style-Attribut über left (Startpunkt) und width (Länge) gesetzt (Listing 3.1 Zeile 5). Erstrecken sich Markierungswidgets über mehrere Zeilen, wird dieses in Einzelwidgets über die betreffenden Zeilen verteilt. Dabei wird die Länge des Widgets der ersten Zeile über Zeile.length - charStart errechnet, alle übrigen Zeilen, die nicht die Endposition enthalten mit Widgets der Länge Zeile.length und dem Startpunkt 0 aufgefüllt und das gesamte Widget durch ein end-Widget mit der Startkoordinate 0 und der Länge Zeile.length-charEnd abgeschlossen. Um zu gewährleisten, dass sich die so fragmentierten Widgetobjekte auf die selbe Markup-Instanz beziehen, werden diese über das Attribut spanID mit dem selben Wert verknüpft. So lassen sich sehr einfach alle fragment-Widgets ermitteln. Dies kann z.B. genutzt werden, um beim Überfahren eines fragment-Widgets alle übrigen Widgets mit einem Highlighting zu versehen (Abbildung 3.4) oder das Widget durch einen Mausklick in einen aktiven Zustand zu versetzen, um weitere Aktionen mit diesem oder dem dazugehörigen Text auszuführen (Einfügen des Textes in die SVG-View beispielsweise). Dies ist zugleich eine Lösung für das Fragmentierungsproblem

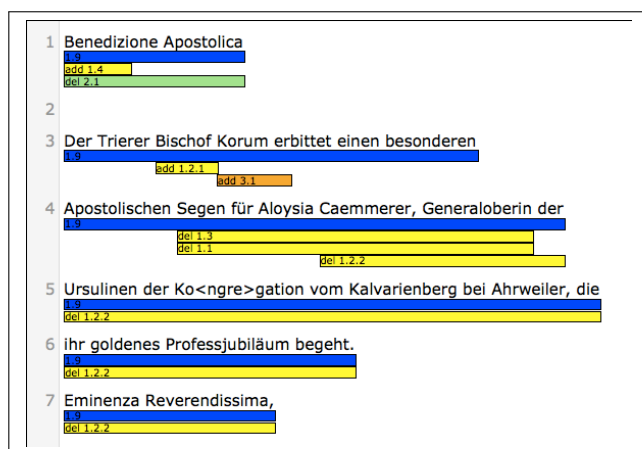


Abbildung 3.4 Linewidgets

der Überlappungen von Markup über Zeilengrenzen hinweg (Abbildung 3.3) Überlappungen stellen in diesem System kein Problem dar, über das sich der Bearbeiter Gedanken machen muss. Sie werden systemintern durch die Kombination aus Zeilenzählung und Zeichenpositionen verwaltet. Die Serialisierung in ein gewünschtes hierarchisches XML-Format kann, um eine genormte Austausch-

barkeit zwischen unterschiedlichen Applikationen zu gewährleisten, mit Hilfe der gespeicherten Positionsinformationen automatisch erzeugt werden.⁴⁰⁹ Neben der genauen Positionierung spielt die Vergabe von Farbattributen eine zentrale Rolle in diesem Konzept. Diese lassen sich während des Transkriptionsprozesses nach Belieben modi-

⁴⁰⁹ Allerdings besteht dabei ein Problem. Da es (bei der TEI zumindest) keine genauen Vorschriften gibt, wie Überlappungen realisiert werden sollen, bleibt immer die Frage bestehen, welche überlappenden Strukturen mit welchen Methoden kodiert werden sollen. Soll prinzipiell eine Fragmentierungstechnik den Vorzug vor einer Milestonetechnik erhalten? Welche Schicht erhält bei der Serialisierung den Vorzug vor den anderen und wird nicht fragmentiert? Soll nur an den Stellen fragmentiert oder die Milestone-Methode verwendet werden, an der sie unbedingt notwendig ist oder grundsätzlich für ganze Schichten? Sollen stand-off-XML-Methoden verwendet werden? Dürfen alle diese Methoden gemischt werden? etc.

fizieren. Somit können innerhalb eines ersten Transkriptionsschrittes die genetischen Informationen einer ersten Bearbeitungsschicht mit unterschiedlichen Farben gekennzeichnet werden. Hier lassen sich unterschiedliche Farbattribute z.B. für Hinzufügungen, Streichungen, Transpositionen oder rdg-Elemente verwenden. Bei der Kodierung einer weiteren Geneseschicht können die Farbwerte der ersten Schicht anschließend über die Konfigurationsdatei neu definiert (z.B. alle weiß) und anschließend zur Verzeichnung einer darauf folgenden Schicht neu vergeben werden. Die Konfigurierbarkeit ist dabei so angelegt, dass sich die Farbwerte an unterschiedliche Kategorien binden lassen, entweder an den Elementnamen selbst (add, del, transpos, rdg etc.) oder an einzelne Attribute, wie z.B. das Untergliederungsschema der einzelnen Geneseschichten. Somit ist neben der Vergabe von Farbattributen für die einzelnen Arten der Modifikationen auch eine farbige Einteilung in Geneseschichten möglich (Schreiber 1 = gelb, Schreiber 2 = grün etc.).

Auch zur Visualisierung der einzelnen Informationsdimensionen der Zeichenketten für die anschließende Publikation kann die Verwendung von Linewidgets sehr sinnvoll sein. So könnte z.B. ein Lesetext generiert werden, der nicht durch Formatierungen oder diakritische Zeichen unterbrochen ist, aber dennoch durch die einblendbaren Linewidgets signalisiert, dass mit der entsprechenden Textstelle zusätzliche Informationen verbunden sind. So hat der Benutzer auf einen Blick genau die Informationen, die er für seine Analyse benötigt. Interessiert er sich nur für eine bestimmte Bearbeitungsschicht oder für eine Art von Modifikationsprozessen, so kann er diese hervorheben. Er kann auch ganze Informationsschichten ausblenden, wie am rdg-Beispiel in Abschnitt 3.4.3 beschrieben wird. Alternativ können die Markup-Strukturen natürlich auch im Sinne einer wysiwyg-Funktion verwendet und ihre Funktion (wenn sie sich sinnvoll durch Textformatierungen darstellen lässt) direkt auf den linearen Text übertragen werden (Abschnitt 3.4.4). Bei räumlichen Informationen sollte allerdings, sofern kodiert, die diplomatische Ausgabe innerhalb der SVG-View bevorzugt werden (Abschnitt 3.3), da die Formatierungen zur Simulation der topographischen Informationen innerhalb des linearen Textes nur zu einer ungenauen Visualisierung führen (vgl. Abschnitt 2.4). Darüber hinaus kann die Darstellung von chronologischen Lesarten aus einer eher prozeduralen analytischen Sicht, wie sie in den Abbildungen in Abschnitt 3.4.4 zu sehen ist, mit Hilfe von Linewidgetelementen sinnvoll ergänzt werden.

Damit zur weiteren Verarbeitung (Verknüpfung mit dem Digitalisat etc.) effizient auf die Informationen eines Widgets zugegriffen werden kann und nicht alle Informationen als Attribute innerhalb der Widgetelemente verwaltet werden müssen, ist mit jedem Widgetelement (bzw. bei mehrzeiligen Widgets mit dem ersten) ein jQuery-Data-Objekt verbunden, das alle textgenetischen Grundinformationen speichert. Das jQuery-Data-Objekt wird für jedes Widget verwaltet und enthält eine Variable namens 'name', in der die Kategorie der Modifikation gespeichert wird (in XML-Notation wäre dies entsprechend der Elementname). Gleich den Attributen der XML-Elemente enthalten diese wiederum spezifische Eigenschaftsvariablen, um die unterschiedlichen Arten von Modifikationen (Hinzufügungen, Entfernungen, Transpositionen und Varianten) eines Textes zu verzeichnen. Folgende textgenetischen Modifi-

kationsprozesse⁴¹⁰ werden in der Basisversion der Anwendung zur Verfügung gestellt (die beschriebenen Kodierungsmöglichkeiten erheben keinen Anspruch auf absolute Vollständigkeit und spiegeln hier lediglich den aktuellen Stand der Kodierungspraxis wider. Das System ist offen für neue Markup-Strukturen. Diese können nach Wunsch einer XML-Struktur hinzugefügt werden, die sich in der Datei `examples.js` befindet. Siehe dazu Abschnitt 3.2.8):

3.2.3 add-Objekt - Hinzufügen von Textsegmenten

Einfaches Einfügen Die Angabe des `@place`-Attributes der TEI kann unterschiedliche Werte annehmen, die vage beschreiben, wo die Einfügung im Original platziert wurde (`below`, `above`, `inline`, etc.). Dieses wird im jQuery-Data-Objekt als Variable `'place'` mit den entsprechenden Werten verwaltet. In vielen Situationen reicht eine solche Beschreibung allerdings nicht aus. Es ist z.B. möglich, dass eine Einfügung zwischen zwei Zeilen beginnt, auf dem Rand der Seite weiter geführt wird und die Einfügung schließlich auf der Rückseite oder einem neuen Blatt endet. Um eine exakte Verortung der Einfügung zu gewährleisten, wird zusätzlich die Variable `'polyID'` zur Verfügung gestellt, die auf ein SVG-Polygon verweist, welches die exakten Koordinaten der Einfügung enthält. Das Anlegen und Verwenden der SVG-Polygone wird in Abschnitt 3.3.2.1 beschrieben. Neben dieser Verknüpfungsmöglichkeit mit dem Digitalisat existiert eine weitere Variante, die nicht nur eine Anzeige des Textausschnitts im Digitalisat ermöglicht, sondern auch ein direktes Überblenden von Text im Digitalisat zulässt (Abschnitt 3.3.3). Diese Variante verweist auf ein komplexes SVG-Textobjekt und wird mit dem dazugehörigen Widgetobjekt über die Variable `'diploID'` verwaltet.

Die Variable `'level'` dient der Zuordnung der Modifikation zu einer bestimmten Geneseschicht. Somit lassen sich die Modifikationen in einer relativen chronologischen Reihenfolge verzeichnen (siehe Abschnitt 2.5.2). Level werden in dieser Notation von Unterlevel durch einen Punkt getrennt (z.B. 1.1.2). Der Wert hinter dem letzten Punkt numeriert die einzelnen Modifikationen innerhalb eines Levels oder Unterlevels. Level stehen nicht nur im `add`-Objekt, sondern auch in allen anderen textgenetischen Objekten zur Verfügung.

Korrektur einer unleserlichen Textstelle durch identischen Text Im vorliegenden Modell wird diese Überschreibung als Einfügung interpretiert und innerhalb des `add`-Objektes durch die Variable `'type'` mit dem Wert `'clarification'` verwaltet. Diese Art der Überschreibung ist zu unterscheiden von der anschließend beschriebenen Variante des Überschreibens, bei der der überschriebene Text nicht durch einen identischen, sondern einen abweichenden Text ersetzt wird.

Einfügung, die gleichzeitig als Streichung dient Bei dieser Art der Einfügung fungiert das ersetzende Zeichen oder Wort als Streichung. Oft passt sich die Überschrei-

⁴¹⁰ Bei der Identifizierung dieser Modifikationsprozesse leistete das TEI-Genese-Modul wertvolle Dienste. Soweit eine entsprechende Kodierung in diesem Modul vorhanden war, richtet sich die Benennung der Variablen und Werte nach diesen Richtlinien. Dies erleichterte unter anderem einen späteren Export der Daten in eine TEI-konforme Ausgabe.

bung dem Schreibfluss des zu tilgenden Zeichens oder Wortes an. Häufig tritt diese Form bei orthographischen oder grammatikalischen Korrekturen auf. Wird z.B. das Wort 'exemplum' durch 'exemplorum' ersetzt, kann dies geschehen, indem das u zu einem o und das m zu einem rum umgeschrieben wird. Solche Veränderungen sind nicht immer leicht zu erkennen. Die Einfügung wird hier als normales add-Objekt realisiert und kann in der Variablen 'function' zusätzlich den Wert 'del' erhalten. Somit kann gekennzeichnet werden, dass die Einfügung gleichzeitig die Funktion einer Tilgung ausübt.

Unmittelbares Einfügen nach einem Tilgungsprozess Bei dieser Art der Tilgung erhält die Variable 'type' den Wert 'instantCorr'. Kann z.B. bei sofort korrigierten Verschreibungen angewendet werden.

Metamarkierungen In den meisten Fällen sind Metamarkierungen zusätzliche Einfügungen (die meist zu einem späteren Zeitpunkt dem Text hinzugefügt werden), weshalb sie hier auch als add-Objekte aufgefasst werden. Metamarks können Angaben für weitere Modifizierungsvorgänge enthalten, wie z.B. Transpositionen oder Angaben über neue Sortierungen von Textelementen. Die Funktion, die diese Metamarkierungen besitzen, werden in der Variable 'function' gespeichert. Diese kann z.B. die Werte 'transposition' oder 'count' (Zählung innerhalb einer neuen Sortierung) annehmen.

Die einzelnen Variablen und Werte der add-Objekte können nach belieben kombiniert werden (so kann z.B. die Sofortkorrektur 'instantCorr' auch gleichzeitig die Funktion 'del' besitzen. Es können auch beliebige weitere Variablen hinzugefügt werden. Gleiches gilt für die folgenden Tilgungsarten.

3.2.4 del-Objekt - Tilgen von Textsegmenten

Seit geraumer Zeit wird in der Editionswissenschaft besonderes Augenmerk auf den Tilgungsprozess und seine Bedeutung innerhalb der Textgenese gelegt. Die mangelnde theoretische Auseinandersetzung mit diesem Phänomen war Anlass der 2009 stattgefundenen Tagung 'Schreiben und Streichen. Zu einem Moment produktiver Negativität in literarischen Schreibprozessen und Textgenesen', die auch unter gleichem Namen als Sammelband erschienen ist. Im Vorwort des Sammelbandes sehen die Autoren den Grund für diese Vernachlässigung darin, „dass sich die traditionellen historisch-kritischen Editionen mit Streichungen vor allem im Hinblick auf die Darstellung von Varianten und Versionen und die Textkritik vor allem im Hinblick auf die Konstitution von Text beschäftigt haben. Darüber hinaus scheint die an positiven Befunden orientierte Textphilologie sich aber auch grundsätzlich schwer zu tun mit der Einbeziehung der Negativität in die Betrachtung des produktiven Schreibens und der Textentstehung.“⁴¹¹ Der „praktisch-produktive Gegenpol zum Fortgang des Schreibens ist nicht

⁴¹¹ Lucas Marco GISI/Hubert THÜRING/Irmgard M. WIRTZ: Schreiben und Streichen. Zu einem Moment produktiver Negativität. Tagungsband, hrsg. v. Irmgard M. WIRTZ, Göttingen, Zürich 2011, S. 8.

das Nicht-Schreiben, sondern das Streichen.“⁴¹² In der „dynamischen Ambivalenz der Streichungen artikuliert sich geradezu das Wesen des Schreibprozesses.“⁴¹³ Die stark von der critique génétique beeinflusste Kategorisierung von Tilgungsprozessen lässt unterschiedliche Einteilungen zu. Grésillon z.B. unterscheidet Tilgungen nach Form, Funktion und Position innerhalb des Textes:⁴¹⁴

Form

1. Die **Tilgung durch eine Streichung**, bei der der 'darunter' liegende Text immer noch mehr oder weniger deutlich lesbar bleibt (wozu auch Schraffierungen oder Schlangenlinien unter oder über dem Text zählen). Hierzu kann das Feld 'state' mit dem Wert 'readable' verwendet werden. Auch wenn bei Grésillon nicht extra erwähnt, sollten zu dieser Kategorie auch alle Zeichen und Symbole gezählt werden, die außerhalb des Textflusses auftreten und eine Anweisung zur Tilgung beinhalten. Diese sollten dann als Metamarkierungen innerhalb des add-Objektes angelegt und mit der Funktion 'del' versehen werden. Eine zusätzliche Variable 'target', die auf die Tilgungspassage verweist, steht zur Verfügung. In ihr wird der level-Wert des jeweiligen add-Elements, das zu tilgen ist, verzeichnet.
2. **Streichungen, die zur Unrekonstruierbarkeit des getilgten Textes führen**. Diese Art der Streichung kann aus intensiven Schraffuren, Radierungen, Auskratzen etc. bestehen. Dies lässt sich mit dem Wert 'unreadable' in 'state' vermerken.
3. Die sogenannte **immaterielle Streichung**, die durch keinen sichtbaren Tilgungsprozess ausgelöst wird, sondern meist nur aus dem Kontext rekonstruiert werden kann, z.B. aus dem „Wiederholungseffekt textähnlicher Sequenzen.“⁴¹⁵ Hierzu kann 'immaterial' innerhalb von 'state' verwendet werden.

Des weiteren wird bei Grésillon eine funktionale Gliederung von Streichungen vorgeschlagen:⁴¹⁶

Funktion

1. Die erste funktionale Kategorie bilden **Streichungen, die zum Zweck der Neuformulierung angebracht wurden**. Hierfür kann in 'function' 'reformulation' angegeben werden.
2. Die zweite beschreibt **Streichungen, die dazu dienen, Text an eine andere Stelle zu verschieben**. (function="transposition")

⁴¹² GISI/THÜRING/WIRTZ: Schreiben und Streichen (wie Anm. 411), S. 8.

⁴¹³ Ebd., S. 9.

⁴¹⁴ GRÉSILLON/ROTHER: Literarische Handschriften (wie Anm. 210), S. 89ff.

⁴¹⁵ Ebd., S. 92.

⁴¹⁶ Ebd., S. 94.

3. Die dritte **Streichungen zum Zweck einer endgültigen Tilgung** (z. B. orthographische oder stilistische Korrekturen, Raffung, Zensur etc.). Dieses kann in der Variable 'function' mit Werten wie 'censorship' vermerkt werden.

Zusätzlich besteht die Möglichkeit, Tilgungen weiter zu kategorisieren. Hierzu kann die Variable type Werte wie 'etch', 'scratch', 'strike' oder ähnliches annehmen. Weitere Unterscheidungen können in der Variable class verwaltet werden, bei strike z.B 'line', 'dotted' etc. Farbattribute können mit 'strikecolor' verzeichnet werden. Hier ist eine Eingabe von RGB-Farbwerten in hexadezimaler Schreibweise möglich (#RRGGBB).

Auch wenn die meisten Schadensbilder, die einen Text betreffen, keine beabsichtigten Tilgungsvorgänge darstellen und durch äußere Einflüsse (Tintenfraß, Wasserschäden etc.) zustande gekommen sind, so lassen sich diese Veränderungen doch auch als Modifikationen auffassen, die das Erscheinungsbild eines Textzeugen zu einem bestimmten Zeitpunkt oder über eine gewisse Zeitspanne verändert haben. Aus diesem Grund werden sie in dieser Anwendung auch als Tilgungselemente definiert. Hierzu kann die Variable cause Werte wie z.B. 'ink' annehmen, um zu kennzeichnen, dass es sich um einen Tintenfleck oder ähnliches handelt.

3.2.5 transpos-Objekt - Transpositionen

Die zuvor beschriebenen Modifikationen beziehen sich vor allem auf Prozesse, die die materiellen Veränderungen einer einzelnen Textüberlieferung betreffen. Aus dieser Perspektive lassen sich Transpositionen auch als kombinierte Prozesse von del und add verstehen. Beim Umstellen eines Textsegments wird es in der Regel getilgt und an einer anderen Stelle eingefügt. Häufig macht es Sinn, einen solchen Prozess mit Hilfe der add- und del-Objekte zu beschreiben, nämlich dann, wenn die Modifikation auch tatsächlich so auf einem Textzeugen verzeichnet vorliegt, also der Text geschrieben, gestrichen und erneut auf dem selben Überlieferungsträger wieder eingefügt wurde.

```

1 <line xml:id="ib3">
2   thi da er du med Himmelen i Pagt; -
3 </line>
4 <line xml:id="ib4">
5   da kan du Folkets Jøkelhjerter tine;
6 </line>
7 <listTranspose>
8   <transpose>
9     <ptr target="#ib4"/>
10    <ptr target="#ib3"/>
11  </transpose>
12 </listTranspose>

```

Listing 3.2

Transpositionen können ihre Wirkung aber auch erst zwischen zwei verschiedenen Textzeugen entfalten, wenn z.B. Metamarkierungen in einem Entwurfstext eine Transposition markieren, die physisch erst in einer späteren Version umgesetzt wurde, also z.B. bei der Abschrift eines Entwurfs zur Erstellung einer Ausfertigung oder beim Überarbeiten von literarischen Entwurfsmanuskripten. Die Metamarkierungen (Pfeile, die vermerken, an welche Stelle ein Textabschnitt verschoben werden soll oder alphanumerische Zählungen, die eine neue Sortierfolge von Textelementen angeben) werden in dieser Anwendung durch add-Objekte beschrieben, die eine Transposition auslösen. Die eigentlichen Textabschnitte, die eine veränderte Position einnehmen, werden mit dem Objekt `transpos` kodiert. Diese Vorgehensweise unterscheidet sich von der der TEI. Die Verzeichnung, die die TEI hier empfiehlt, ist mit einem hohen Kodierungsaufwand verbunden. Zuerst müssen alle Textsegmente, die an der Transposition beteiligt sind, markiert und in einem separaten XML-Fragment in eine andere Reihenfolge gebracht bzw. ihre veränderte Position angegeben werden (Listing 3.2).⁴¹⁷

Austauschen von Elementen In vorliegendem System werden hingegen beide Zeilen als `transpos`-Objekte markiert (Abbildung 3.5), wobei das zweite lediglich als Ankerobjekt dient (`type=anchor`) und das erste mit `type=after` und `target=1.1` angibt, dass dieses zu einem späteren Zeitpunkt (Level=2.1) hinter das Ankerobjekt mit Level=1.1 platziert

```

2 ...
3 thi da er du med Himmelen i Pagt; —
  transpos 2.1 type:after target:1.1
4 da kan du Folkets Jøkelhjerter tine;
  transpos 1.1 type:anchor
5 ...

```

Abbildung 3.5

wird. Für die Variable `type` stehen vier unterschiedliche Werte zur Verfügung: ‘anchor’, ‘before’, ‘after’ und ‘replace’. Der Anker bezeichnet ein Element, das selbst nicht von einer Umstellung betroffen ist und nur als Einfügeposition dient. Im vorliegenden Beispiel hätte alternativ auch das erste Textsegment als Anker und das zweite mit den

entsprechenden Transpositionskoordinaten versehen werden können. ‘After’ und ‘before’ geben an, ob das zu verschiebende Element hinter dem oder vor dem Ankerelement eingefügt wird. ‘Replace’ kann in den Fällen verwendet werden, wo ein Textsegment verschoben und dafür gleichzeitig ein anderes Element ersetzt wird. Sollte sich zwischen den beiden Textzeilen noch weiterer Text befinden, der nicht von der Transposition betroffen ist, kann dies so kodiert werden,

```

3 thi da er du med Himmelen i Pagt; —
  transpos 1.1 type:replace target:2.1
4 ...
5 ...
6 ...
7 da kan du Folkets Jøkelhjerter tine;
  transpos 2.1 type:replace target:1.1
8 ...

```

Abbildung 3.6

wie in Abbildung 3.6 zu sehen (Textsegment 1 ersetzt Textsegment 2 und gleichzeitig wird Textsegment 1 von Textsegment 2 ersetzt).

⁴¹⁷ Um die Metamarkierungen verkürztes Beispiel entnommen aus <http://www.tei-c.org/release/doc/tei-p5-doc/de/html/PH.html#transpo>

Komplexere Umstellungen - Reihenfolge verändern (besonders bei Listeneinträgen) Analog zu letzterem Beispiel können auch komplexere Modifikationen verzeichnet werden, wie in Abbildung 3.7 zu sehen (vereinfachtes Beispiel ohne Metamarkierungen). Dabei ersetzen sich Textsegmente nicht gegenseitig, sondern die Textsegmente an den Positionen, an denen sie später in der neuen Sortierung erscheinen sollen. Diese Variante verzeichnet nun allerdings nicht, zu welchem Zeitpunkt die Transposition stattgefunden hat. Um ein vollständiges Beispiel in diesem Falle zu geben, hätten sich die einzelnen transpos-Widgets auf zuvor mit add-Widgets markierte Textsegmente beziehen müssen. Damit wäre dann auch die relative Abfolge der Transpositionen kodierbar gewesen (die add-Widgets hätten dann mit den Levelstufen der Form 1.x und die Transpositionslevel der Überarbeitungsstufe mit Werten der Form 2.x versehen werden müssen).

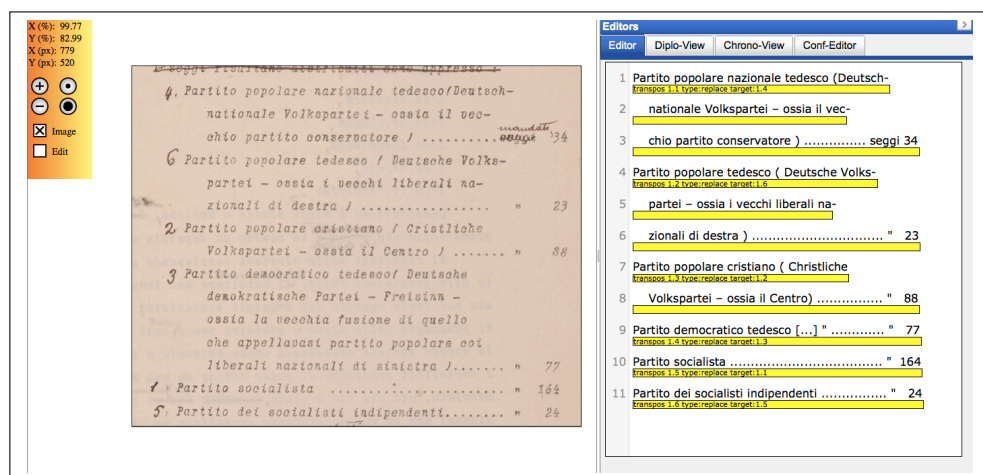


Abbildung 3.7

3.2.6 Modifikationen wiederholen oder zurücknehmen

Einfügungen oder Tilgungen von Textsegmenten können wiederholt werden, um einen Sachverhalt zu bekräftigen, z.B. wiederholtes Streichen oder nochmaliges Überschreiben des Textes mit der gleichen Zeichenkette. Im Gegensatz zu der im add-Abschnitt beschriebenen 'Korrektur einer unleserlichen Textstelle durch identischen Text' (clarification) handelt es sich hierbei zwar um einen identischen Modifikationsprozess, aber mit einer anderen Intention. Da dieser Vorgang theoretisch häufiger zu unterschiedlichen Zeitpunkten wiederholt werden kann, sollte er in solchen Fällen mit einer weiteren Widgetmarkierung (add oder del) und dem entsprechenden Level-Wert (wann dies geschehen ist) versehen werden. Alternativ kann auch die Variable 'redo' verwendet werden. Mit 'redoLevel' kann der Zeitpunkt der Rücknahme kodiert werden. Sollten Prozesse wieder zurückgenommen werden, geschieht dies in der Regel durch Metamarkierungen (z.B. Schlangenlinien unter einer Unterstreichung zum Signalisie-

ren, dass diese zurückgenommen wird). Dies kann mit der Variable ‘undo’ verzeichnet werden, der Zeitpunkt dementsprechend mit ‘undoLevel’.

3.2.7 rdg-Objekt - Variantenverzeichnung unterschiedlicher Textzeugen

Bei der Variantenverzeichnung unterschiedlicher Textzeugen wird das rdg-Objekt verwendet. Dieses enthält eine Variable mit Namen ‘source’, in der die jeweilige Handschrift notiert werden kann. Verzeichnete rdg-Objekte können wiederum alle zuvor beschriebenen Objekte enthalten, da die Textsegmente der jeweiligen Handschrift wiederum aus komplex ausgezeichneten genetischen Prozessen bestehen können (zur Verwendung und zu den Besonderheiten dieses Objektes siehe Abschnitt 3.2.10).

3.2.8 Anlegen von neuen Markup-Elementen und Dateneingabe

Neben den in den letzten Abschnitten beschriebenen Konfigurationsmöglichkeiten für textgenetische Prozesse werden noch eine Reihe weiterer Objekte und Variablen benötigt, die die Positionierung, Darstellung und Verwaltung der mit diesen Objekten verbundenen Widget-Objekte betreffen. Über die Eigenschaft ‘type’ wird das Widgetobjekt näher bestimmt. Mögliche Werte sind ‘start’, ‘end’, ‘both’ oder ‘fragment’. Widgets, die sich über mehrere Zeilen erstrecken, bestehen aus Einzelwidgets. Das erste Widget, das den Startpunkt einer Textmarkierung beschreibt, erhält den Wert ‘start’, alle folgenden Widgets, die keine Endposition markieren, sich also über die ganze Zeile erstrecken, den Wert ‘fragment’ und Widgets mit der Endposition der Textmarkierung den Wert ‘end’. Befindet sich die Endposition der markierten Textpassage in derselben Zeile wie die Startposition, wird dies durch die Eigenschaft ‘both’ angegeben. Alle Fragment- und End-Widgets werden über die spanID mit dem start-Widget verbunden. Alle spanIDs eines mehrzeiligen Widgets enthalten dabei die eindeutige id des start-Widgets. Diese setzt sich zusammen aus der Zeilennummerierung und einer counter-Variable, die für jede Zeile die vorhandenen Widgets zählt (‘line_’+lineCount+‘_’+widgetCount). Alle start- oder both-Widgets enthalten zusätzlich Angaben darüber, in welcher Zeile (‘lineStart’ und ‘lineEnd’) und an welcher Zeichenposition (‘charStart’ und ‘charEnd’) sie beginnen und enden. Über die Eigenschaften polyID und diploID können die Widgets mit SVG-Objekten der SVG-View verknüpft werden. Dies können einfache Polygone sein, die die markierten Textpassagen des Editortextes mit den zugehörigen Koordinaten des Digitalisats verbinden oder komplexere SVG-Textobjekte, die den markierten Text direkt ins Digitalisat einbinden.

Jedes start- und both-Widget enthält die zuvor beschriebenen textgenetischen Eigenschaften in Form eines jQuery-Data-Objektes, das seine Informationen aus den Nutzereingaben bezieht. Dazu wird aus der de_elements-Liste, die sich in der Datei examples.js befindet, für jedes Textgenese-Objekt ein Eingabemenü generiert. Für das transpos-Objekt sieht der Eintrag in dieser aus XML-Fragmenten bestehenden Liste folgendermaßen aus:⁴¹⁸

⁴¹⁸ Im Prinzip könnten nach diesem Muster für alle Elemente der TEI Elementdeklarationen ange-


```

1 <element name="transpos">
2   <attr name="type">
3     <val>anchor</val>
4     <val>after</val>
5     <val>before</val>
6     <val>replace</val>
7   </attr>
8   <attr name="target">
9     <val>textfield</val>
10  </attr>
11 </element>

```

Listing 3.3

Alle Attribut-Einträge eines Elements werden in HTML-Auswahllisten transformiert, die die vordefinierten Werte der entsprechenden value-Einträge in Auswahlmenüs übersetzen. Ist 'textfield' in value definiert, wird stattdessen ein input-Feld generiert, das eine freie Texteingabe zulässt. Diese Formularmasken werden im Eingabemenü Text-Markup zur Verfügung gestellt (Abbildung 3.9). Um chronologische Abfolgen und alternative Lesarten und Bündelungen unterschiedlicher Modifikationen zu ermöglichen, sind unter jedem Eingabeblock zwei Level-Felder notiert. Nach dem in Abschnitt 2.5.2 beschriebenen Gliederungsformat lassen sich die Modifikationsprozesse in Level und Sublevel unterteilen. Um diese Zählung nicht immer manuell eintragen zu müssen, stehen zwei Felder zur Verfügung. Im ersten kann ein Level frei vergeben werden, im zweiten wird der zuletzt verwendete um eins erhöht (1.1 auf 1.2). Gewöhnt man sich beim Auszeichnen der Texte an, die Modifikationen nach einer ersten Basisreihenfolge zu notieren, kann diese Methode die aufwendigere manuelle Eingabe der Level-Werte ersetzen.

3.2.9 add- und rootAdd-Objekt

Texteinfügungen werden in der Anwendung entweder mit add- oder rootAdd-Objekt kodiert. Die Grundidee der add-Variante besteht darin, dass die Usability eines Bearbeitungsvorganges gesteigert werden kann, wenn analog zu den nicht-linearen Schreibräumen auch deren Bearbeitungsräume voneinander getrennt werden. Bei der Verwendung von add wird der einzufügende Text nicht direkt in der Haupteditorinstanz verwaltet, sondern in einer separat angelegten. Dies ist vor allem dann sinnvoll, wenn Texte aus komplexen Geneseschichten bestehen, wie im Musil-Beispiel aus Abbildung 2.16. Die grundlegende Funktionsweise ist in Abbildung 3.8 zu sehen.

legt werden. Somit könnte die Editorkomponente auch als Online-TEI-Editor genutzt werden. Evtl. wäre hier auch zu überlegen, einen automatischen Export aus einem beliebigen TEI-Schema zu implementieren.

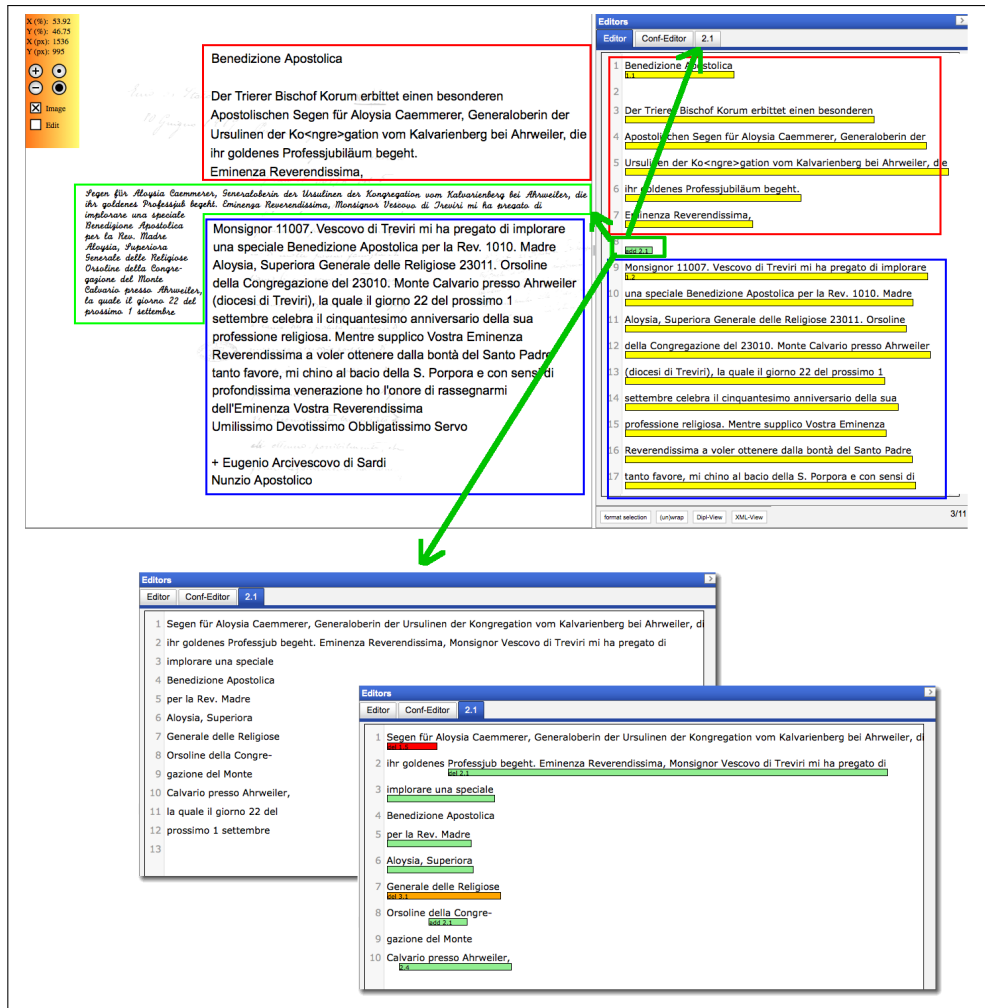


Abbildung 3.8

In vorliegendem fiktiven Beispiel sind die beiden Textsegmente der ersten Geneschicht eines Schreibers X in zwei Fragmente eingeteilt worden (rotes und blaues Kästchen). Der Text der beiden Fragmente wurde im Haupteditor eingefügt und mit rootAdd markiert. Anhand dieser Markierungen konnte der Text topographisch genau in das Digitalisat eingefügt werden (siehe Abschnitt 3.3.3). Das Textsegment, das in einem späteren Überarbeitungsprozess von einem Schreiber Y umfließend um das zweite Textsegment eingefügt wurde, ist hingegen nicht direkt im Haupteditor zwischen den beiden Fragmenten platziert worden. Statt dessen wurde mit Hilfe von add lediglich eine Markierung an dieser Stelle angebracht, dadurch automatisch ein weiteres Editorfenster angelegt und in die Editorreiterleiste eingefügt. Das add-Widget des Haupteditors beschreibt in diesem Fall nicht eine Information des darüberlie-

genden Textes, sondern markiert nur eine Art Ankerpunkt im Text. In der linearen Abfolge befindet sich der Text virtuell an der Startposition dieses Anker-Widgets. Das so separierte Textsegment kann wiederum komplexe genetische Informationen enthalten (zweites Editorfenster unter der Hauptansicht), die nun separat in der neuen Editorinstanz bearbeitet werden können. Somit lassen sich also alle Hinzufügungen einer Geneseschicht sehr übersichtlich in einzelnen Editorinstanzen verwalten. Die Zugehörigkeit zu einer Schicht wird dabei über die Namen der Editorinstanzen signalisiert. Diese bestehen aus den für die add-Objekte verwendeten Level. Wenn also mehrere topographisch verteilt vorliegende Einfügungen einer Geneseschicht in diesen separierten Editorinstanzen verwaltet werden, können alle Schichten leicht analysiert werden, indem nacheinander alle Reiter mit dem jeweiligen übergeordneten Level geöffnet werden (1.1, 1.2, 1.3 etc.). Diese Architektur könnte darüber hinaus auch als Darstellungsmodus für die Onlinepublikation der Edition dienen. So wären die einzelnen Bearbeitungsschichten visuell sauber getrennt und ließen sich über die Ankerpunkte dynamisch an der jeweiligen Stelle oder in einem separaten Bereich einblenden. Allerdings darf nicht unerwähnt bleiben, dass die Separierungsmöglichkeiten nur die Hinzufügungen betreffen. Schichtenbezogene Tilgungen lassen sich auf diese Art nicht ohne weiteres separieren, da sie ja unmittelbar zwei unterschiedliche Schichten betreffen können (Schreiber Y streicht Text von Schreiber X). Diese können dann nur visuell getrennt werden, indem etwa alle Streichungen des Schreibers X in der diplomatischen Ansicht oder im linearen Text hervorgehoben werden. Ein zweites Problem besteht darin, dass es mehr als eine Ankerposition im linearen Text geben kann. Betrachtet man z.B. eine abweichende Rezeptions- und Produktionschronologie, so wären mehrere Anker im Text zu platzieren, die zu unterschiedlich angeordneten Textsegmenten führen würden. Alternativ zu der Ankerpunkt-Methode besteht die Möglichkeit, abweichende Abfolgen außerhalb des eigentlichen Transkriptionstextes zu verwalten (siehe dazu Abschnitt 3.4.1).

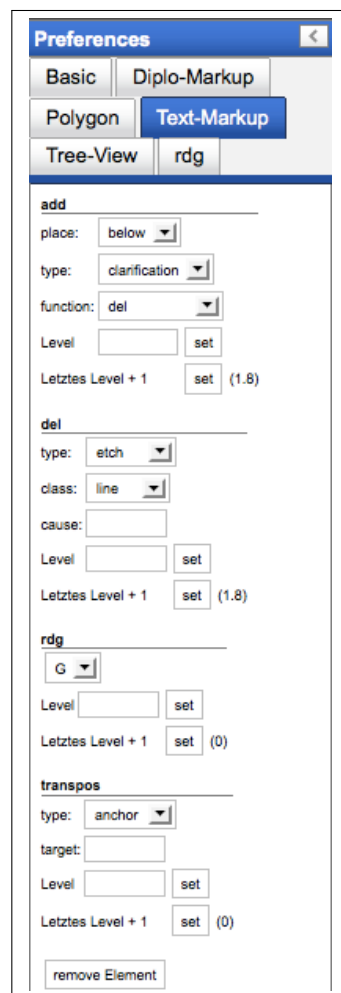


Abbildung 3.9

3.2.10 Besonderheiten des rdg-Objekts

Mit dem rdg-Objekt lassen sich Varianten unterschiedlicher Textträger verzeichnen. Hierfür werden in der Konfigurationsdatei für jeden Textzeugen Widgettypen definiert. Das Beispiel in Abbildung 3.11 hat folgende Konfigurationsdefinition:

```

1 <tagname value="rdg" dispTagname="true">
2   <item name="G">background-color:yellow;</item>
3   <item name="D">background-color:lightgreen;</item>
4   <item name="g">background-color:orange;</item>
5   <item name="d">background-color:lightgrey;</item>
6 </tagname>

```

Listing 3.4

Die rdg-Widgets (so wie alle anderen Widgets auch) können mit css-Eigenschaften formatiert werden. Die Angaben dazu werden innerhalb der item-Elemente notiert. In diesem Beispiel wurden verschiedene Farben gewählt, um die Handschriften-Widgets zusätzlich zu ihrer separaten Linewidgetposition visuell voneinander abzuheben.



Abbildung 3.10

Die name-Attribute der item-Elemente werden aus der Konfigurationsdatei in das Bearbeitungsmenü Text-Markup übernommen und dem Benutzer als Auswahlliste zur Verfügung gestellt (Abbildung 3.10). Ein ähnliches Konzept, wie das der multiplen Editorinstanzen der add-Objekte unterstützt die Verwaltung der einzelnen rdg-Objekte. Der Haupteditor kann dazu dienen, im Sinne einer Leithandschrift oder einer fiktiven Rekonstruktion (z.B. zum Erstellen eines Archetypen), einen Basistext zu

notieren oder die Textzeugen parallel zu verzeichnen, ohne eine besondere Handschrift zu begünstigen. Im letzteren Fall wird im Haupteditor kein rekonstruierter Text erfasst, sondern der einer beliebigen Handschrift.

Ob die Edition eine gleichwertige Sicht auf die varianten Textzeugen bietet oder eine bestimmte Überlieferung bevorzugen möchte, hängt in erster Linie von der Methode ab, mit der die rdg-Widgets positioniert werden (später mehr dazu). Nach dem Markieren einer Textstelle und dem Auswählen einer Handschrift aus der Auswahlliste öffnet sich ein Editorfenster, in dem die jeweilige Variante eingetragen wird. Zusätzlich wird, wie bei den add-Objekten in Abschnitt 3.2.9 schon beschrieben, eine neue Editorinstanz erstellt, um evtl. weitere Auszeichnungen an der varianten Textpassage vornehmen zu können. Bei dem Beispiel in Abbildung 3.11 handelt es sich um eine Strophe aus Wolfram von Eschenbachs Parzival. Dieses Beispiel wurde schon in Abschnitt 2.1.1 verwendet, um den Gebrauch des klassischen lemmatisierten Einzelstellenapparats zu demonstrieren.

Die Widgets erstrecken sich über das Textsegment eines Basistextes, zu dem eine Variante in der entsprechenden Handschrift existiert. Sie dienen in diesem Sinne als Lemma-Markierung. Die freien Räume der Handschriftenzeilen, die von keinem Widget belegt werden, sind dabei gedanklich mit dem Basistext aufzufüllen. Sind die varianten Zeichenketten länger als die des markierten Basistextes (Abbildung 3.11, Zeile 5 g: fiktive Erweiterung des Textes), passen sie also nicht komplett in das rdg-Widget, wird der

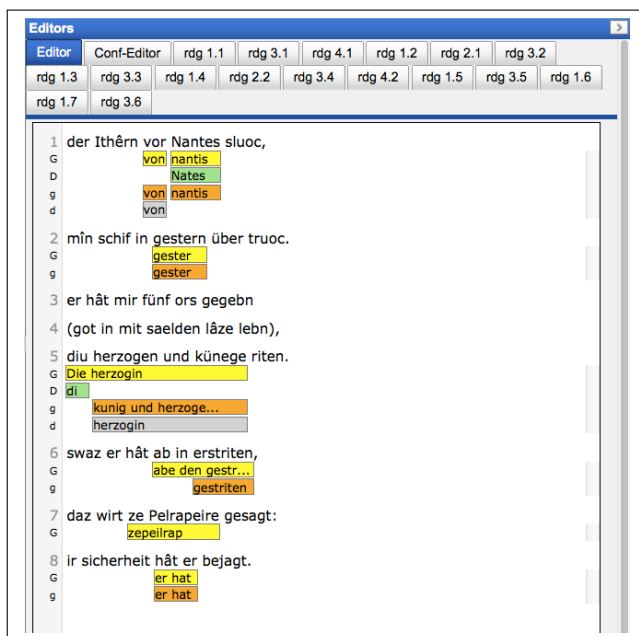


Abbildung 3.11

Text an geeigneter Stelle abgebrochen und durch drei Punkte signalisiert, dass sich eine längere Variante dahinter verbirgt. Diese kann über einen maussensitiven Tooltip eingeblendet werden (Abbildung 3.12, links).

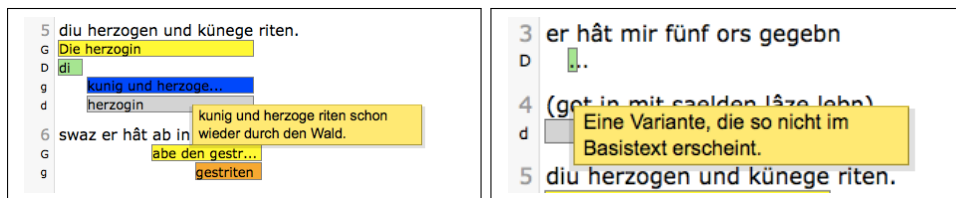


Abbildung 3.12

Das Widgetprinzip ermöglicht zudem, Textpassagen zu verzeichnen, die keine Entsprechung im Basistext haben. Hierzu wird an der gewünschten Cursorposition ein Widget eingefügt, das nur aus drei Punkten besteht. Der Text kann hier wieder durch ein Popupfenster eingeblendet werden (Abbildung 3.12, rechts).

Soll der umgekehrte Fall verzeichnet werden (in einem varianten Textzeugen befindet sich kein Gegenstück zu einer Textpassage des Basistextes), so wird an dieser Stelle ein 'leeres' Widget angelegt. Es enthält dann keinen Text und keine Punkte, jedoch die Farbe der jeweiligen Handschrift und somit ist auf den ersten Blick ersichtlich (Abbildung 3.13, links), dass sich in besagter Handschrift eine Überlieferungslücke befindet. Bei der hier beschriebenen Form der Variantenverzeichnung handelt es sich

auf den ersten Blick um eine lemmatisierte Darstellung in Bezug auf einen Basistext. Diese Basistextnotation hat den Vorteil, dass Widgets sich beliebig überlappen dürfen, da sie ja nur die Varianten zum Basistext bilden.

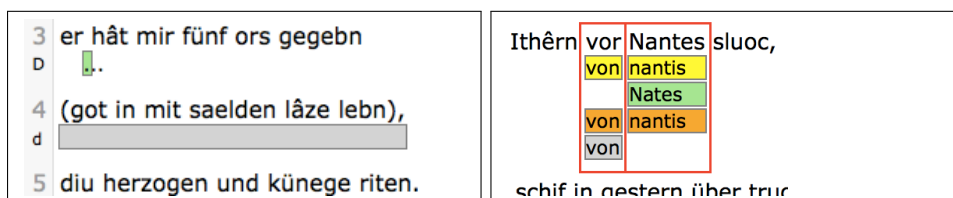


Abbildung 3.13

Je nachdem, wie die Widgets platziert werden, kann das rdg-Widget aber auch dazu verwendet werden, den Text im Sinne der *parallel-segmentation method* der TEI so zu verzeichnen, dass alle Varianten als gleichwertig und austauschbar betrachtet werden können. Hierzu muss nur darauf geachtet werden, dass die Widgets, die parallele Varianten darstellen sollen, untereinander Textblöcke bilden, sie also an den gleichen Zeichenpositionen eingefügt werden müssen (Abbildung 3.13, rechts, entspricht Zeile 1 aus Abbildung 3.11). Nach dieser Notation kann aus jeder beliebigen Handschrift ein Basistext rekonstruiert und anstelle des Textes im Haupteditor eingesetzt werden. Dazu wäre jeweils der variante Text in der Editortextzeile des Basistextes an den start- und end-Koordinaten des entsprechenden Widgets durch den Text im auszu-tauschenden Handschriftenwidget zu ersetzen und umgekehrt der variante Basistext in das nun frei gewordene Widget einzufügen. Bei vielen parallel verzeichneten Varianten könnte es zudem sinnvoll sein, die Reihenfolge, in der die Varianten angezeigt werden, nach unterschiedlichen Kriterien zu sortieren und so Gruppen von ähnlichen Varianten zu bilden. So könnten z.B. auf Grundlage von Ähnlichkeitsalgorithmen regional bedingte Schreibvarianten von Orts- oder Personennamen (die ja vor allem in mittelalterlichen Quellen noch keiner festen Normierung unterlagen) identifiziert und daraus gegebenenfalls auch Rückschlüsse auf den Entstehungsort der Handschrift gezogen werden. Dies wäre neben der Verwendung von phylogenetischen Algorithmen eine weitere Möglichkeit, Ähnlichkeitsbeziehungen zwischen Textzeugen zu ermitteln.

Der Bearbeitungsmodus eignet sich auch zur Weiterbearbeitung von automatisch erzeugten Kollationierungsdateien, die wie in Abschnitt 1.2.4.3 schon beschrieben, nicht immer zu den gewünschten Ergebnissen führen. Dafür wäre lediglich ein Stylesheet oder Exportskript zu schreiben, das aus dem TEI-Exportformat von Juxta oder CollateX das hier verwendete stand-off-Format generiert (zum Datenmodell dieser Anwendung siehe Abschnitt 3.2.12, zu einem Beispiel eines umgekehrten Serialisierungsprozesse von dem hier verwendeten stand-off-Format in die double end-point attachment method-Variante der TEI siehe Abschnitt 3.2.13).

3.2.11 Konfigurationsmöglichkeiten

Bei einer browserbasierten Editions Umgebung ist das Medium, in welchem die Edition erstellt wird das gleiche, in dem sie anschließend auch publiziert wird. Daraus ergeben sich wesentliche Vorteile gegenüber einer Trennung dieser beiden Ebenen in einen Erstellungsprozess mit Hilfe eines nicht-browserbasierten Editionswerkzeugs in Form einer Desktopanwendung oder einer Kodierung mit XML-Editoren und der darauf folgenden browserbasierten WWW-Publikation. Zum einen können so alle Komponenten, die für die Bearbeitungssoftware entwickelt wurden, in gleicher oder nur geringfügig angepasster Form auch für die Publikationsumgebung genutzt werden, genau wie es ja auch schon seit längerem gefordert wird: dem Benutzer einer Edition die gleichen Möglichkeiten zur Verfügung zu stellen, die auch der Editor bei ihrer Erstellung hatte.⁴¹⁹ Neben den Kontextmaterialien selbst schließt dies auch alle analytischen Instrumente, die bei der Transkription zum Einsatz kamen, ein. Damit würde die Digitale Edition ihren Schwerpunkt auch viel weiter in Richtung der schon seit längerem angemahnten Berücksichtigung des Anwenderinteresses (Stichwort *user-driven*)⁴²⁰ verschieben, als es in herkömmlichen Editionen bisher der Fall war. Der Nutzer ist somit selbst Wissenschaftler und kann eigene Interpretationen oder Sichten produzieren und diese zur Diskussion stellen.⁴²¹ Zum anderen erlaubt eine solche Vorgehensweise im Sinne eines *wysiwyg*-Prinzips, die Edition so zu entwickeln, wie sie später einmal aussehen soll. Das betrifft nicht nur die typographisch gestaltete Ansicht des linearen Editionstextes, sondern ebenso die Darstellung des Digitalisats und der Navigations-, Recherche und Analysekomponenten. Darüber hinaus können aus einer browserbasierten Anwendung heraus leichter unterschiedliche Sichten auf den Editionstext, seine Kontextmaterialien und die Digitalisate erzeugt werden. Diese können dann sowohl als eine Art Vorschlag des Projektteams verstanden werden, sich aber auch selbst vom Nutzer erzeugen und mit anderen Interessenten oder der Allgemeinheit teilen lassen. Was dazu benötigt wird ist ein Mechanismus, der in der Lage ist, verschiedene Einstellungen von Ansichten zu speichern und die Editions-/Publikationssoftware mit diesen Einstellungen zu laden. Bezogen auf die hier vorgestellte Applikation sind mögliche Anwendungsfelder von solchen speicherbaren Konfigurationseinstellungen folgende:

1. Veränderung der visuellen Darstellung von Markup-Widgets während eines Bearbeitungsprozesses.
2. Speicherung von unterschiedlichen Sichten:
 - Bildbearbeitungseinstellungen (Helligkeit, Kontrast, Farbwertkorrektur)

⁴¹⁹ „Der Umgang des Benutzers mit dem Text ähnelt aber u.U. den Fragestellungen des Editors, so dass ersterem eigentlich die gleiche Arbeitsumgebung zur Verfügung gestellt werden müsste wie letzterem.“ SAHLE: *Digitale Editionsformen 2* (wie Anm. 2), S. 61. Vgl. auch Ulfert RICKLEFS: *Zur Systematik historisch-kritischer Ausgaben*, in: *Editio 13* 1999, S. 1–22.

⁴²⁰ REHBEIN: *Reflections* (wie Anm. 397).

⁴²¹ Vgl. hierzu schon Ross über die Rolle des Rezipienten als Produzenten: „The future of editing lies in restructuring textuality on screen. In the near future hypertext will lower the barriers between writers, readers, and editors, turning all into true collaborators in the writing of Textuality.“ Charles L. ROSS: *A future for Editing: Lawrence in Hypertext*, in: *Textual studies and the common reader: essays on editing novels and novelists 2000*, S. 141–159.

- Visualisierung der Linewidgets (Farben, Ein- oder Ausblenden von Informationsschichten)
- Bildausschnitte und verknüpfte Digitalisate
- Unterschiedliche Chronologien und Lesarten

3. 'Zitieren' von Text- und Bildausschnitten.

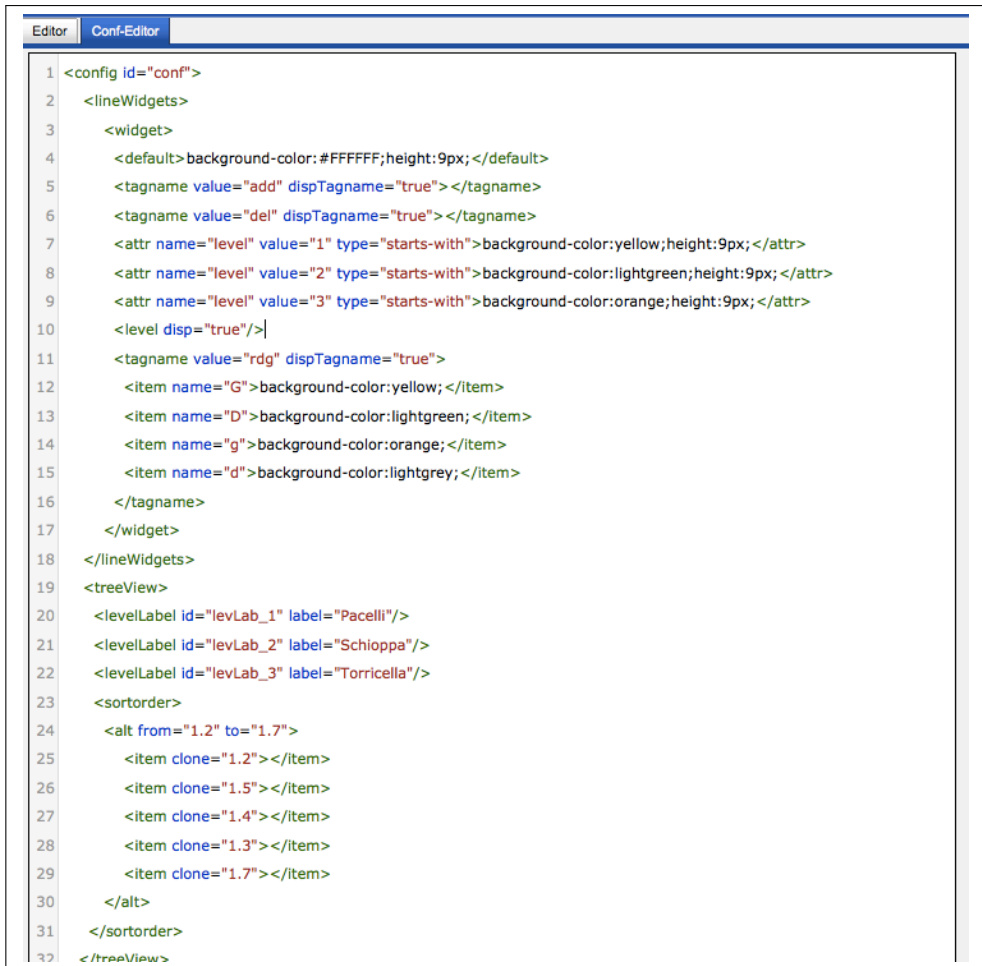
3.2.11.1 Darstellung von Markup-Widgets

Diese Art der visuellen Konfigurierbarkeit unterstützt den Transkriptionsprozess. Wird die Bearbeitung in unterschiedliche Phasen eingeteilt, kann so die Übersichtlichkeit der für diese Phasen relevanten Markup-Strukturen gesteigert werden. Dies hat eine einfachere und effizientere Kodierungspraxis zur Folge. Hierzu dient die Konfigurationsdatei, die in der Editormenüleiste unter dem Reiter Conf-Editor zu finden ist. Veränderungen an dieser Datei während eines Bearbeitungsvorganges wirken sich unmittelbar auf das Erscheinungsbild bzw. auf die weitere Eingabe (z.B. von Widgetelementen) aus. Darüber hinaus können die so konfigurierten Einstellungen dazu dienen, serverseitig als Grundeinstellungen für die Ansicht eines Textzeugen oder für Gruppen von Textzeugen zu fungieren. Abbildung 3.14 zeigt einen Ausschnitt der Konfigurationsmöglichkeiten.

Im widget-Abschnitt können Einstellungen für die Darstellung der einzelnen Markup-Elemente definiert werden. In diesem Beispiel beschreibt `<default>` das Aussehen eines Widgets, für das keine spezifischeren Konfigurationen vorgenommen wurden (weiß mit der Höhe von 9 Pixeln). Mit `<tagname value="add" dispTagname="true"/>` wird festgelegt, dass in allen add-Widgets der Typ (add) angezeigt wird und mit `<attr name="level" value="1" type="starts-with">`, dass alle Widgets, die mit einem Genese-Level versehen wurden, der mit 1 beginnt (`type="starts-with"`), die css-Formatierung `'background-color:yellow;height9px'` erhalten. Damit lassen sich verschiedene chronologische Geneseschichten unterschiedlich visualisieren. Darüber hinaus können auch zusätzliche Formatierungsanweisungen für die Sublevel definiert werden. Hierfür steht das Attribut `type="starts-with"` zur Verfügung. So würde z.B. `<attr name="level" value="1.3" type="starts-with">` das Aussehen aller Widgets beeinflussen, die sich innerhalb des Sublevelbereichs 1.3 befinden. Theoretisch ließen sich diese Einstellungen auch auf die SVG-Objekte der diplomatischen Ansichten (Abschnitt 3.3) übertragen. Somit können verschiedenen Geneseschichten (z.B. unterschiedlichen Schreibern) eigene Text- und Streichfarben zugeordnet werden. (Der darauf folgende Konfigurationsabschnitt für die rdg-Widgets wurde bereits in Abschnitt 3.2.9 beschrieben.)

3.2.11.2 Speicherung unterschiedlicher Sichten

Wie eingangs beschrieben, kann in einer browserbasierten kollaborativen Editions Umgebung die Konfigurationsdatei unterschiedliche Zwecke erfüllen. So könnte sie auch dazu dienen, alle aktuellen Einstellungen der Anwendung zu speichern, z.B. Bildausschnitte der SVG-View samt eingblendeter Genesestufen, alternativer Verzeichnun-



```

1 <config id="conf">
2   <lineWidgets>
3     <widget>
4       <default>background-color:#FFFFFF;height:9px;</default>
5       <tagname value="add" dispTagName="true"></tagname>
6       <tagname value="del" dispTagName="true"></tagname>
7       <attr name="level" value="1" type="starts-with">background-color:yellow;height:9px;</attr>
8       <attr name="level" value="2" type="starts-with">background-color:lightgreen;height:9px;</attr>
9       <attr name="level" value="3" type="starts-with">background-color:orange;height:9px;</attr>
10      <level disp="true"/>|
11      <tagname value="rdg" dispTagName="true">
12        <item name="G">background-color:yellow;</item>
13        <item name="D">background-color:lightgreen;</item>
14        <item name="g">background-color:orange;</item>
15        <item name="d">background-color:lightgrey;</item>
16      </tagname>
17    </widget>
18  </lineWidgets>
19  <treeView>
20    <levelLabel id="levLab_1" label="Pacelli"/>
21    <levelLabel id="levLab_2" label="Schioppa"/>
22    <levelLabel id="levLab_3" label="Torricella"/>
23    <sortorder>
24      <alt from="1.2" to="1.7">
25        <item clone="1.2"></item>
26        <item clone="1.5"></item>
27        <item clone="1.4"></item>
28        <item clone="1.3"></item>
29        <item clone="1.7"></item>
30      </alt>
31    </sortorder>
32  </treeView>

```

Abbildung 3.14

gen, Bündelungen textgenetischer Modifikationsprozesse oder aktivierter Widgeteinstellungen, quasi ein ‘Einfrieren’ aller Komponentenzustände. Damit wiederum ließen sich unterschiedliche Sichten auf das Dokument serverseitig speichern und in Diskussionsforen z.B. dazu verwenden, zu einer bestimmten Textstelle oder einem vergrößerten Bildausschnitt einen Link auf die jeweilige Konfigurationsdatei zu generieren. Beim Aufruf des Links würde die Anwendung dann mit den entsprechenden Einstellungsparametern geladen. Dazu müssten die einzelnen Konfigurationsdateien lediglich personalisiert und in einer eigenen Benutzer-Config-Verwaltung abgelegt werden.

An den Widgetabschnitt schließen sich die Konfigurationsmöglichkeiten für die chronologischen Lesarten an (vgl. auch Abschnitt 3.4.1). Hier können z.B. Labels für die Baumansicht der genetischen Abfolge von Modifikationsprozessen definiert werden. Im sortorder-Abschnitt darunter werden die zusätzlichen alternativen Rei-

henfolgen, Lesarten und Bündelungen von Prozessen verwaltet. Je nachdem welche Lesart hier für eine Sicht bevorzugt wird, könnte anhand der Sortierfolge ein linearer Text aufgebaut werden, der genau dieser Reihenfolge entspricht, oder es könnten in der diplomatischen Ansicht (sofern diese für den Textzeugen angelegt wurde) die entsprechenden Elemente nacheinander eingeblendet werden.

Auch die Bilddarstellungsoptionen lassen sich aus der Konfigurationsdatei ermitteln und separat speichern. Wie eingangs beschrieben, ergibt sich ein wesentlicher Vorteil einer browserbasierten Arbeitsumgebung aus der Tatsache, dass der Bearbeiter der Edition direkt mit dem Werkzeug arbeitet, das auch dem Nutzer später zur Verfügung steht. Er ist nun z.B. nicht mehr gezwungen, Images eines Textzeugen mehrfach zu bearbeiten und die verschiedenen Versionen auf dem Server zu hinterlegen, damit diese bei Bedarf aufgerufen werden können. Solche Versionen entstehen z.B. dann, wenn unterschiedliche Bildbereiche unterschiedlicher Filterfunktionen bedürfen, um ihre Lesbarkeit zu verbessern oder z.B. die Konturen von Wasserzeichen durch Negativdarstellungen hervorzuheben (Abbildung 3.15). Dies kann schnell zu einer Vielzahl unterschiedlicher Image-Dateien führen. In einer Anwendung, wie der hier vorgestellten, kann das Digitalisat in seinem Originalzustand verbleiben und stattdessen die Filterfunktionen aus dem Basic-Menü der Anwendung verwendet werden, um nicht das Image selbst sondern die Einstellungsparameter zu speichern (also eine Art Snapshot zu generieren). Dies könnte so funktionieren, dass in der Konfigurationsdatei für verschiedene Einstellungen XML-Blöcke mit den jeweiligen Filter-Parametern definiert und diese zusammen mit dem Digitalisat im jeweiligen DBMS verwaltet würden (z.B. eine Basis- und weitere alternative Bearbeitungseinstellungen). Beim Aufruf der Editionsseite würde dann das Original in die SVG-View geladen und mit den entsprechenden Werten der Basiseinstellungen initialisiert werden. Der Nutzer könnte je nach Verwendungszweck aus der Liste der gespeicherten Konfigurationen eine alternative auswählen oder selbst die Parameter mit Hilfe der Schieberegler modifizieren.

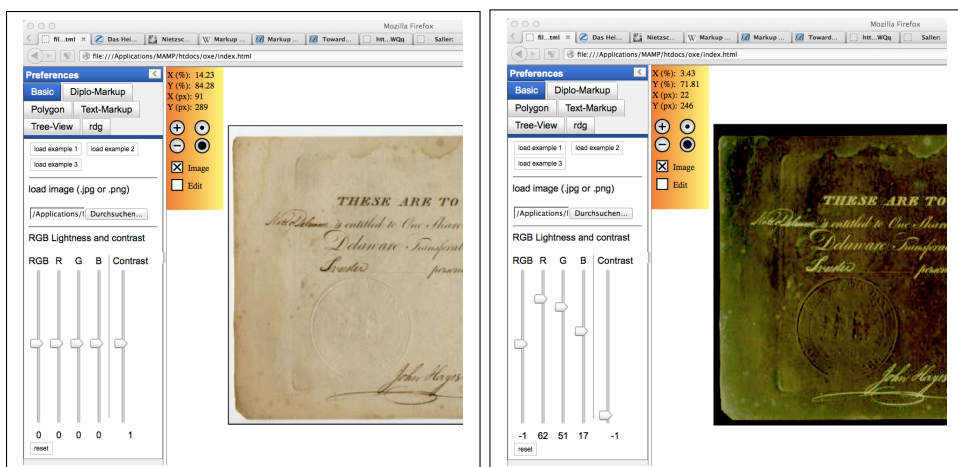


Abbildung 3.15 Wasserzeichen

3.2.11.3 'Zitieren' von Text- und Bildausschnitten

In textgenetischen Digitalen Editionen besteht eine besondere Herausforderung darin, die Zitierfähigkeit von Texten zu gewährleisten, die sich gerade durch ihre dynamischen Veränderungsprozesse auszeichnen.⁴²² Wie schon in Abschnitt 2.5.1 beschrieben, handelt es sich bei diesen Textzeugen oft um Entwurfsstufen, die so gesehen nur Momentaufnahmen innerhalb einer größeren Textgenese darstellen. Die Frage stellt sich also, wie diese 'flüchtigen' Texte zitiert werden können, wenn nicht bloß eine Endversion oder eine fiktiv rekonstruierte Version des Textes zitiert werden soll. Zeilenbasierte Verfahren scheinen nicht wirklich sinnvoll zu sein. Zum einen können sich diese innerhalb der Textgenese leicht verändern, zum anderen sind sie schon auf dem eigentlichen Textzeugen nicht leicht zu identifizieren, handelt es sich ja oft eher um Schreibräume, die erst bei der Transkription in eine linearisierte Reihenfolge gebracht werden. Zusätzlich sind verschiedene Lesarten dieser Reihenfolgen zu berücksichtigen. Eine Zitierung nach einer produktionschronologischen Rekonstruktion würde eine andere Reihenfolge der Textsegmente ergeben, als eine rezeptionschronologische, und damit auch verschiedene Zeilenzählungen. In Editionen, die ein so großes Gewicht auf nicht-lineare und nicht fixierte Texte legen, wie es bei textgenetischen Editionen in der Regel der Fall ist, müssen also auch die Zitiermechanismen flexibler werden.

Wenn aber zeilenbasierte Mechanismen für lineare Texte nicht mehr funktionieren und man sich zudem so genau wie möglich auf das Original beziehen möchte, kann als zitierfähiges Objekt eigentlich nur eine Einheit auf der Ebene von mit Markup versehenen Textsegmenten oder ein definierter Bildausschnitt des Digitalisats in Frage kommen. Zusätzlich wäre ein Mechanismus zu implementieren, der den Benutzer eines Zitierlinks direkt an die gewünschte Position leitet, indem z.B. das entsprechende Widget oder auch der Text direkt gehighlightet werden oder der Imageviewer den entsprechenden Bildausschnitt anzeigt. Hierzu könnten unterschiedliche Varianten dienen:

1. Mit Leveln gekennzeichnete Markup-Widgets
2. Beliebig auswählbare Textsegmente
3. Mit Polygonen markierte Bildausschnitte

Die einfachste Variante im Bereich der Zitierung eines Textausschnittes bestünde darin, die in der Basistranskription ausgezeichneten Textsegmente zu verwenden. Durch die eindeutige Levelnummerierung ließe sich beim Aufruf des Zitierlinks entweder das entsprechende Markup-Widget oder die dazugehörige Textstelle highlighten. Sollte mit dem Widget auch ein Bildausschnitt in Form eines SVG-Polygons verknüpft sein, könnte zugleich auch dieser hervorgehoben oder vergrößert werden. Im Prinzip der gleiche Mechanismus ließe sich auch für die zweite Variante, die einer beliebigen Textauswahl, implementieren, wenn dazu die Funktionalität der Editorinstanzen genutzt wird. Hier wird im Gegensatz zur ersten Variante nicht eine schon existierende

⁴²² Bernhard ASSMANN: Sind die kanonischen Zitierweisen der Geisteswissenschaften als nachhaltige Komponenten digitaler Repositorien geeignet? (Magisterarbeit), Universität zu Köln 2005, URL: <http://www.cei.lmu.de/pub/MagArbAssmann.pdf>, Vgl.

Markup-Struktur verwendet, sondern eine weitere externe angelegt. Dabei könnte der Editor (in einem non-editable-Modus) genutzt werden, um die Koordinaten der selektierten Textstelle zu ermitteln und daraus einen Link zu erzeugen, der alle Informationen enthält, um bei einem Laden der Anwendung ein (Zitier-)Widget anzulegen und an der entsprechenden Position einzufügen oder ein direktes Highlighting der Textstelle zu bewirken. Hieran zeigt sich erneut der Vorteil einer modularisierten Trennung von interpretativem stand-off-Markup und Basistranskription. Es muss nicht mehr zwangsläufig ein fixierter statischer Text referenziert werden. Was referenziert wird, ist ein Textfragment innerhalb einer möglichen Anordnung von Textsegmenten. Der Nachteil dieser Variante besteht darin, dass nach einem solchen Zitieren entweder keine Änderungen mehr am Text vorgenommen werden dürfen, oder der Zitierlink an diese Textversion gebunden werden muss, also gleichzeitig das System dafür Sorge zu tragen hat, dass diese Version weiterhin verfügbar bleibt. Allerdings besteht dieses Problem ebenso bei einer zeilen- oder gar bei einer dokumentbasierten Zitierweise, da ja nicht mit Sicherheit davon ausgegangen werden kann, dass sich eine Veränderung des Dokumenttextes nicht auch auf den durch das Zitat gekennzeichneten Sachverhalt auswirkt. Wesentlich stabilere Links in Bezug auf ihr Referenzobjekt können durch Koordinaten beschriebene Bildausschnitte bieten. Zwar sind sie an das jeweilige Digitalisat gebunden, aber in der Regel ändert sich die Anordnung der Textsegmente auf dem Original nicht, so dass dadurch ein relativ zukunftssicheres Zitiersystem etabliert werden könnte. Die Transkription wäre dann nicht mehr der eigentlich zu zitierende Text, sondern nur noch eine mögliche Realisierung in Form eines temporär existierenden linearen Textes. In einer Anwendung wie der hier vorgestellten könnte ein solcher Link einfach über die Koordinaten eines in der SVG-View angelegten Polygons generiert werden. Leider kann diese Variante nur dann verwendet werden, wenn auch das Digitalisat zur Verfügung steht. Außerdem besteht keine automatische Verbindung zwischen Bildausschnitt und Transkription. Diese müsste über ein Widget in der Transkription erst wieder erzeugt werden. Für die drei beschriebenen Varianten könnten so folgende Links verwendet werden:

1. `transkrX/levelY`
2. `transkrX/textfragmentY/linestart=...&lineend=...&charstart=...&charend=...`
3. `imageX/polygon[d=....]`

Ob sich eine Zitierpraxis anhand solcher Methoden als stabil genug erweist, bleibt vorerst abzuwarten, da hierfür allgemeinere Kriterien gefunden werden müssten. Es kann z.B. nicht Sinn eines Zitierlinks sein, dass er nur in einem ganz bestimmten Softwarekontext funktioniert. Insofern scheint eine Zitierung anhand des Originals oder besser anhand der Koordinaten des Digitalisats erstrebenswerter.

3.2.12 Datenformat

Textgenetische Transkriptionen enthalten Datenstrukturen, die alle Probleme multipler Hierarchien beinhalten: Seien es einfache überlappende Strukturen (wie beim

klassischen linebreak-Fragmentierungsproblem) oder diskontinuierliche Fragmentierungen,⁴²³ wie beispielsweise bei der Kombination von Textfragmenten, um unterschiedliche chronologische Abläufe zu kodieren (Abschnitt 2.5.2). Hierbei handelt es sich auch nicht bloß um rein theoretisch existierende Probleme, wie Cummings bemerkt.⁴²⁴ Die Notwendigkeit, alle diese Strukturen auch in der Praxis kodieren können zu müssen sind so schon von Bird und Liberman erkannt worden.⁴²⁵ Es ist nicht ersichtlich, warum eine textgenetische Edition zugunsten einer spezifischen Informati-onsschicht (sei es nun die physische Dokumentstruktur der diplomatischen Transkription oder irgendeine von vielen denkbaren dieser zuwiderlaufenden logischen Strukturen) eine mit ihr konkurrierende vernachlässigen sollte.⁴²⁶ Eine einmal getroffene Entscheidung für eine Sichtweise würde die Kodierungsmöglichkeiten der anderen Sichten zu stark einschränken. Analytische Prozesse auf diese Daten müssten entweder in redundanten Strukturen vorgehalten werden, um die inhaltslosen Milestone-Lösungen oder die unzulänglichen Fragmentierungen zu umgehen (vgl. Abschnitt 2.7), oder die nicht bevorzugte Hierarchie müsste dynamisch zuvor in eine Hierarchie 'ersten Ranges' transformiert werden. Dies hätte allerdings zur Folge, das langwierige Transformationsprozesse notwendig und nur im Anschluss an diese weitere Analyseprozesse ausführbar wären, die ihrerseits wieder auf weitere (Vor-)Verarbeitungsprozesse angewiesen sein könnten, wie z.B. das Indexieren der Daten, um Suchanfragen auszuführen. Dies ist der Hauptgrund, warum in der vorliegenden Anwendung eine stand-off-Markup-Struktur gewählt wurde, die im Prinzip beliebig viele Sichten gleichwertig kodierbar macht. Das schon angesprochene Datenformat, in dem die Transkriptionen und Digitalisatinformationen gespeichert werden, ist ein XML-stand-off-Format, das die kodierten Informationen über Zeilen und Zeichenpositionen verwaltet. Listing 3.5 zeigt Beispiele für add-, del- und rdg-Widgets. In `<rootText>` ist der Text des Haupteditorfensters verzeichnet. Texte selbst besitzen kein eingebettetes Markup, dieses wird durch separate Widget-Elemente kodiert. Die Widgets enthalten eine eindeutige id aus Zeilennummer und Widgetzähler, den Namen des Elements, den

⁴²³ Zum Problem der 'discontinuous structures', also verteilten Fragmentierungen, die nicht unmittelbar aufeinander folgen siehe C. M. SPERBERG-MCQUEEN/Claus HUITFELDT: Markup Discontinued: Discontinuity in TexMecs, Goddag structures, and rabbit/duck grammars, in: Balisage Series on Markup Technologies 1 2008, URL: <http://www.balisage.net/Proceedings/vol1/html/Sperberg-McQueen01/BalisageVol1-Sperberg-McQueen01.html>.

⁴²⁴ „However, there is more theoretical interest in the use of forms of boundary marking or stand-off markup, or various non-XML solutions.“ CUMMINGS: Companion (wie Anm. 84), §25.

⁴²⁵ Intervalle mit Lücken, Sequenzen von überlappenden Intervallen etc, von denen die Hierarchische Struktur nach dem OHCO-Modell nur eine mögliche von vielen darstellt. Steven BIRD/Mark LIBERMAN: A Formal Framework for Linguistic Annotation, in: Linguistic Data Consortium 2000, URL: <http://arxiv.org/abs/cs/0010033>, S. 13.

⁴²⁶ Auch wenn dies in der Praxis oft geschieht: „the majority of projects using TEI get along just fine without using complicated solutions for resolving these conflicts. Why is this? Simply, they are content to prioritize one hierarchy over the other. In most cases they have the creation of an end resource in mind, which will be easier with one of these hierarchies and simply refers to the other as an extra source of information. For example, most projects encode the intellectual structure of the document and simply record its physical manifestation as milestone references, but if their end product needs individual pages they would choose to prioritize this hierarchy instead.“ CUMMINGS: Companion (wie Anm. 84), §25.

Geneselevel, die Zeilen und Zeichenpositionen und evtl. weitere textgenetische Informationen.

```

1 <rootText>
2   Der Trierer Bischof Korum erbittet einen besonderen\n
3   Apostolischen Segen für Aloysia Caemmerer, Generaloberin der\n
4   Ursulinen der Kongregation vom Kalvarienberg bei Ahrweiler, die ...\n
5 </rootText>
6 <widget name="add" id="line_2_1" level="1.2.1" lineFrom="2" lineTo="2"
   charStart="12" charEnd="12">
7   <text>Benedizione Apostolica\n
8     per la Rev. Madre\n
9     Aloysia, Superiora\n
10    Generale delle Religiose\n
11    Orsoline della Congre-\n
12    gazione del Monte\n
13    Calvario presso Ahrweiler,\n
14    la quale il giorno 22 del\n
15    prossimo 1 settembre</text>
16   <g id="polyCan_2" widgetID="line_2_1">
17     <path id="path_2" d="M240,724 L89,1404"/>
18   </g>
19   <g id="PolyText_2" inclination="rotate" x="270" y="734" transform="rotate
     (12.519885912351455, 240, 724)" dxVal="0" wSpace="0" font-family="
     Pilgi" font-size="60"/>
20 </widget>
21 <widget name="del" parent="root" id="line_3_0" level="1.3" lineFrom="3" lineTo
   ="3" charStart="14" charEnd="56" type="strike" class="line" color="#008000
   "/>
22 <widget name="rdg" parent="root" id="line_0_1" value="G" level="1.1" lineFrom=
   ="0" lineTo="0" charStart="11" charEnd="14">
23   <text>Nates</text>
24 </widget>

```

Listing 3.5

Das Widget in Zeile 6 ist nach dem in Abschnitt 3.2.9 beschriebenen Konzept der separierten Editorinstanzen in den Text der Haupteditorinstanz eingefügt worden (deswegen beschreiben die line- und char-Koordinaten keinen Textrange sondern einen Einfügepunkt: Zeile 2, Zeichenposition 12). Innerhalb des Widgets ist in <text> der an dieser Stelle eingefügte Text gespeichert. Sollte der Text weitere Markup-Strukturen enthalten, so werden diese innerhalb weiterer Widgets kodiert, die sich auf der gleichen Hierarchieebene befinden wie <text>, also Kindelemente des add-Widgets sind. Das SVG-g-Element mit der id polyCan_2 (Zeile 16) speichert über path die Polygon-Koordinaten (d-Attribut), anhand derer der Text in die diplomati-

sche Ansicht eingefügt wird. Im zweiten g-Element (Zeile 19) werden die Formatierungsanweisungen für dieses diplomatische Textsegment verwaltet. In diesem Fall die Art der Transformation (*inclination=rotate*), die Winkelangabe der Rotation (*transform*), Zeichenabstand (*dxVal*), Wortabstand (*wSpace*), Schrifttyp und Schriftgröße. Das Widget in Zeile 21 ist ein Beispiel für eine grün linierte Streichung (*name="del"*, *type="strike"*, *class="line"*, *color="#008000"*), die sich über ein Textsegment des Haupteditors von Zeichenposition 14-56 in Zeile 3 erstreckt. Darunter ist ein Beispiel für ein *rdg*-Widget mit den entsprechenden Zeilen- und Zeichenpositionen und dem Namen der Handschrift (*value*) definiert. In `<text>` befindet sich der Variantentext.

3.2.13 Serialisierung

Das in Abschnitt 3.2.12 beschriebene Datenformat enthält alle Informationen, um unterschiedliche Exportformate zu generieren. Dies können nach den Richtlinien der TEI gültige stand-off-Formate mit XInclude und xpointer-Techniken sein⁴²⁷ oder eingebettete, OHCO-basierte Markup-Strukturen. Allerdings ist selbst bei einer Serialisierung in ein valides TEI stand-off-Format Vorsicht geboten, da es hierbei zu Komplikationen bei der Kodierung überlappender Hierarchien kommen kann. Es ist zwar möglich, das komplette Markup eines Dokumentes zu externalisieren und den Text über XInclude-Statements zu referenzieren, allerdings muss das Ergebnis der Internalisierung wieder aus gültigem XML bestehen. Insofern wird das Problem nur in die stand-off-Dokumente verlagert, wie folgendes Beispiel zeigt:

```

1 <rootText>Der Trierer Bischof Korum erbittet einen besonderen\n
2 Apostolischen Segen für Aloysia Caemmerer, Generaloberin der\n
3 Ursulinen der Kongregation vom Kalvarienberg bei Ahrweiler, die\n
4 </rootText>
5 <widget name="del" parent="root" id="line_3_0" level="1.3" lineFrom="1" lineTo
   = "3" charStart="14" charEnd="9" type="strike" class="line" color="#008000"
   />

```

Listing 3.6

Das Widget beschreibt eine einfache Streichung, die sich über drei Zeilen erstreckt. Eine nach dem von der TEI empfohlenen stand-off-Mechanismus komplett externalisierte Kodierung ist in Listing 3.7 zu sehen. Zum einen ergibt sich so eine sehr komplexe, unhandliche Struktur innerhalb der stand-off-Dokumente, zum anderen müssen erneut Fragmentierungs- oder Milestone-Techniken verwendet werden, um aus dem externalisierten Code wieder gültiges XML zu generieren. Ein weiteres Problem bei einer Serialisierung in ein TEI-Format lässt sich gut am Beispiel des *apparatus*-Modul der TEI verdeutlichen. Auf das in Abschnitt 2.6 schon beschriebene Problem der unzureichenden Standardisierung einer einzigen *apparatus*-Methode soll an dieser Stelle

⁴²⁷ <http://www.tei-c.org/release/doc/tei-p5-doc/en/html/SA.html#SASOso>

```

1  <!-- irgendwo im Textbody von Source.xml -->
2  <content>Der Trierer Bischof Korum erbittet einen besonderen\n
3  Apostolischen Segen für Aloysia Caemmerer, Generaloberin der\n
4  Ursulinen der Kongregation vom Kalvarienberg bei Ahrweiler, die\n
5  ...
6  </content>
7
8  <!-- externalisiertes stand-off-Dokument (Fragmentierungs-Variante): -->
9  <l>
10   <xi:include href="Source.xml" xpointer="string-range(element(/1),0,13)"/>
11   <del>
12     <xi:include href="Source.xml" xpointer="string-range(element(/1),14,51)"/
13   >
14   </del>
15 </l>
16 <l>
17   <del>
18     <xi:include href="Source.xml" xpointer="string-range(element(/1),52,111)"/
19   >
20   </del>
21 </l>
22 <l>
23   <del>
24     <xi:include href="Source.xml" xpointer="string-range(element(/1),112,121)"/
25   >
26   </del>
27   <xi:include href="Source.xml" xpointer="string-range(element(/1),122,165)"/>
28 </l>
29
30 <!-- oder in der Milestone-Variante: -->
31 <l>
32   <xi:include href="Source.xml" xpointer="string-range(element(/1),0,13)"/>
33   <del xml:id="level1.3_1" target="#level1.3_2"/>
34   <xi:include href="Source.xml" xpointer="string-range(element(/1),14,51)"/>
35 </l>
36 <l>
37   <xi:include href="Source.xml" xpointer="string-range(element(/1),52,111)"/>
38 </l>
39 <l>
40   <xi:include href="Source.xml" xpointer="string-range(element(/1),112,121)"/>
41   <del xml:id="level1.3_2"/>
42   <xi:include href="Source.xml" xpointer="string-range(element(/1),122,165)"/>
43 </l>

```

Listing 3.7

noch einmal genauer eingegangen werden. Die einzige brauchbare Form der Verzeichnung von Varianten mit Hilfe der TEI scheint die *double end-point attachment method* zu sein, da sie allein die Möglichkeit bietet, die gesamte Funktionalität aller übrigen Methoden abzubilden. Es ist nicht genau ersichtlich, warum die TEI darüber hinaus noch weitere Methoden zulässt. Die Hauptgründe scheinen, wie schon erwähnt, darin zu liegen, dass dies zum einen eine Konzession an die klassische Editionspraxis in Printeditionen ist und zum anderen diese Methoden angeboten werden, um den Kodierungsaufwand in Grenzen zu halten. Hier liegt aber das eigentliche Problem. Funktional gesehen sind die Methoden nicht identisch. Entweder lassen sich keine Überlappungen (*parallel-segmentation method*) modellieren oder die Methode führt zu unzureichenden Lemmamarkierungen, die keine genaue Referenz auf den Basistext erlauben (*location-referenced method*). Es ist nicht zielführend, dass sich ein technisches Datenformat unterschiedlichen Rahmenbedingungen anpasst, die für die Verarbeitungslogik auf Softwareebene hinderlich sind. Aus Sicht der Applikation ist die Speicherstruktur am sinnvollsten, die alle denkbaren auftretenden Kodierungssituationen abdeckt, also funktional die meisten Möglichkeiten bietet. Es ist nicht Aufgabe eines Standards, weniger funktionale Konzepte zuzulassen, nur um die Lesbarkeit und eine einfachere Bearbeitung zu gewährleisten oder sich einer spezifischen Editionspraxis anzupassen (zumal wenn diese noch aus der Druckkultur stammt und aufgrund völlig anderer medialer Voraussetzungen entstanden ist). Vielmehr ist es Aufgabe der Applikation selbst, dafür zu sorgen, dass die Bearbeitung möglichst wenig Aufwand bedeutet und am Ende eine eindeutige Kodierung zustande kommt. Sollten spezifischere Datenstrukturen in einem bestimmten Anwendungskontext benötigt werden, um z.B. die klassische Darstellung eines externen Apparates außerhalb der Transkription zu produzieren (*location-referenced method* im *external-Modus*) oder um die Varianten direkt innerhalb der Transkription mit Tooltips oder ähnlichen Techniken zu visualisieren (*location-referenced method* im *internal-Modus*), dann ist es Aufgabe der Applikation, aus der Grundkodierung diese spezifische Ausgabeform zu erzeugen. In Softwaresystemen müssten so auch keine unterschiedlichen Transformationsmechanismen mehr implementiert werden, nur um aus den verschiedenen Methoden erst die aus Datenstruktursicht logischste Form zu generieren. Sollten z.B. Datensammlungen aus unterschiedlichen Quellen stammen, die variierende Methoden der Kodierung verwendet haben, so ist das ein mühsamer und überflüssiger Prozess, der bei einer eindeutigen Kodierung nicht hätte implementiert werden müssen.

Aus dem in dieser Arbeit verwendeten Datenformat ließen sich im Prinzip alle unterschiedlichen Kodierungsvarianten der TEI-Apparate erzeugen. Allerdings scheint es sinnvoller zu sein, nur die *double end-point attachment method* als gültigen Serialisierungsstandard zu implementieren, da sich alle anderen Apparatypen aus ihr ableiten lassen.⁴²⁸ Auch die Angabe des lemma-Elements innerhalb dieser Kodierung ist unnötig, da die Informationen schon über die *from*- und *to*-Koordinaten gespeichert werden. Eine zusätzliche Angabe führt hier nur zu redundanter Datenspeicherung.

⁴²⁸ Dass der reine Kodierungsaufwand hier höher ist, als bei den anderen Methoden, fällt nicht ins Gewicht, da sie ja nicht manuell kodiert werden muss, sondern erst bei der Serialisierung erzeugt wird.

Das jeweilige Lemma sollte lediglich dann aus den Koordinaten rekonstruiert werden, wenn es für einen bestimmten Output benötigt wird, also z.B. bei der Ausgabe eines traditionellen lemmatisierten Apparates für eine Druckversion. Für das rdg-Modul ist eine Serialisierung in Form der double end-point attachment method implementiert worden:

```

1 <l xml:id="WBP.7" n="7">
2   daz <anchor xml:id="WBP-A6.1"/>wirt <anchor xml:id="WBP-A6.2"/>ze <anchor
      xml:id="WBP-A6.3"/>Pelrap<anchor xml:id="WBP-A6.4"/>eire<anchor xml:id
      ="WBP-A6.5"/> gesagt:<anchor xml:id="WBP-A6.6"/>
3   <app from="#WBP-A6.2" to="#WBP-A6.5">
4     <rdg wit="#G">zepeilrap</rdg>
5   </app>
6   <app from="#WBP-A6.1" to="#WBP-A6.4">
7     <rdg wit="#D">Test1</rdg>
8   </app>
9   <app from="#WBP-A6.3" to="#WBP-A6.6">
10    <rdg wit="#g">Test2</rdg>
11  </app>
12 </l>

```

Listing 3.8 Serialisierung nach der double end-point attachment method

In Beispiel 2 kann diese über die Schaltfläche XML-View generiert werden. In einer Server-Client-Architektur kann die dahinter liegende Funktion `serialize("XML")` dafür genutzt werden, das so erzeugte TEI-Dokument weiter zu verarbeiten (z.B. um es in einem DBMS zu speichern).

3.3 SVG-View

Bei der Transkription handelt es sich um einen recodierenden Prozess von dokument-bezogenen Informationen. Nach Pichler besteht Sinn und Zweck einer Transkription in der exakten Übertragung aller Informationen eines Dokumentes, Textes etc. mit dem Ziel einer möglichst korrekten Repräsentation des Originals.⁴²⁹ Nicht unproblematisch sind natürlich Terminologien wie die der objektiven Wiedergabe und die des identischen Textes und die Frage nach dem Gegenstand, was und auf welcher Ebene transkribiert werden soll. Der sprachliche Ausdruck als graphematisch und orthographisch normalisierte, 'gesicherte' Textwiedergabe? Oder doch die genaue layoutwiedergebende diplomatische Transkription?⁴³⁰ Wie inzwischen allgemein anerkannt ist, handelt es sich bei der Informationskodierung von materiellen Befunden nie um ein

⁴²⁹ PICHLER: Transcriptions (wie Anm. 31).

⁴³⁰ Diese Problematik beschreibt Sahle ausführlich. SAHLE: Digitale Editionsformen 3 (wie Anm. 54), 251ff.

objektives Verfahren, da jede Interpretation eines Befundes einen Prozess der Informationsdeutung voraussetzt.⁴³¹ Auch der Leser spielt hier eine nicht zu unterschätzende Rolle. Der Editor verfolgt mit der Transkription Ziele, die eng mit der Erwartungshaltung seines Publikums verbunden sind und zudem häufig durch disziplinspezifische Editionspraktiken und -theorien beeinflusst werden. Der Transkriptor (Editor) spielt dabei eine Mittlerrolle zwischen der Aussage und Bedeutung des Textes selbst in seinem spezifischen (historischen) Kontext und dem Publikum. Diese Art der Informationsübersetzung besteht immer aus einem zweigliedrigen Prozess: zum einen aus der Decodierung zwischen Original und Editor und zum anderen aus der Recodierung zwischen Editor und Leser. Dem Editor obliegt die verantwortungsvolle Aufgabe, dem Leser den 'Code von historisch Vergangenen' in einen zeitgemäßen 'lesbaren' Code zu übersetzen. Die Entwicklungen in der Editionswissenschaft führen schon seit längerem weg von dem Anspruch, die Autorintention zu 'übersetzen', vielmehr wird die Authentizität der Quelle in den Mittelpunkt gestellt.⁴³² Nach Sahle kann „Diese Authentizität aber [...] allein den physisch vorhandenen Dokumenten der Überlieferung“⁴³³ zukommen. Das heißt dann aber auch, dass Transkription und Original (Digitalisat) in einen engeren Zusammenhang gestellt werden müssen, nicht zuletzt deswegen, um die Authentizität der Transkription am Original überprüfen zu können. Wie schon zuvor erwähnt, kann dabei weder die Transkription vollständig durch das Faksimile ersetzt werden, noch umgekehrt. Nach wie vor ist das Zielpublikum auf das Expertenwissen der Editoren angewiesen (kodikologisches, epigraphisches, papyrologisches, philologisches, linguistisches, historisches Fachwissen etc.) und erst durch ihre Sachkenntnis wird der Leser in die Lage versetzt, die Informationen zu deuten und zu bewerten. Und andersherum wird genau an diesem Punkt die prinzipielle Bedeutung der Faksimile klar, weil nur anhand dieser der Leser die Möglichkeit hat, die Entscheidungen des Editors nachzuvollziehen und gegebenenfalls zu korrigieren und neu zu bewerten. Erst in einem System von enger gegenseitiger Referenzierung von transkribierter Information und materiellem Befund kann von einer wirklichen überprüfbar Authentizität gesprochen werden. Allerdings ist es bisweilen sehr mühselig, ohne solche Verknüpfungstechniken die Verbindung zwischen Transkription und Faksimile herzustellen, vor allem dann, wenn es sich um schwer rekonstruierbare oder komplex strukturierte Überlieferungen handelt. Nicht nur für den Laien ist es oft schwierig, den transkribierten Text auf dem Digitalisat zu lokalisieren. In den Beispielen textgenetischer Editionen wird dies augenscheinlich.

Aktuell gibt es verschiedene Konzepte, wie in solchen Editionen versucht wird, diese Verbindungen deutlich zu machen: zum einen mit Verlinkungsmechanismen, die den Text sowohl in der Transkription als auch im Faksimile parallel hervorheben, zum anderen mit dem Versuch, die Texttopographie in Form (ultra-)diplomatischer Abschriften so genau wie möglich zu simulieren. Eine dritte, bis heute noch gar nicht

⁴³¹ ZELLER: Befund und Deutung. Interpretation und Dokumentation als Ziel und Methode der Edition (wie Anm. 187).

⁴³² Dirk van HULLE: Authenticity or Hyperreality in Hypertext Editions. Notes Towards a Searchable 'Recherche', in: Human IT 1. Tidskrift för studier av IT ur ett humanvetenskapligt perspektiv 1999, URL: etjanst.hb.se/bhs/ith/1-99/dvh.htm.

⁴³³ SAHLE: Digitale Editionsformen 2 (wie Anm. 2), S. 139.

genutzte Möglichkeit besteht darin, den Text direkt in das Faksimile zu integrieren. Dies setzt allerdings eine so exakte diplomatische Abschrift voraus, dass in browserbasierten Umgebungen die HTML-Methoden unweigerlich an ihre Grenzen stoßen (vgl. Abschnitt 2.4). Neben den ersten beiden Methoden soll in dieser SVG-basierten Anwendung besonderes Augenmerk auf die Möglichkeiten von Überblendungstechniken gelegt und manuelle und (semi)automatische Verfahren evaluiert werden, mit deren Hilfe solche ultradiplomatischen Transkriptionen realisiert werden können.

Zur Zeit existieren zwar einige Systeme, die es erlauben, Bild und Textinformationen miteinander zu verknüpfen (auch unter dem Namen Text-Image-Coupling oder Linking bekannt⁴³⁴), aber keines erfüllt die spezifischen Anforderungen eines browserbasierten Werkzeugs, das neben der Verknüpfung auch die Integration des Textes in Form von diplomatischen Transkriptionen in das digitale Faksimile unterstützt. So ist z.B. das am Maryland Institute for Technology und an der Indiana University entwickelte browserbasierte Text-Imagelinking-Tool TILE (Text-Image Linking Environment)⁴³⁵ zwar in der Lage, solche Verknüpfungen zu erstellen, es unterstützt den Import und Export von transkribiertem Text aber nur auf Zeilenbasis. Die Verknüpfung mit Bildinformationen wird über Rechtecke gelöst, die aus HTML-div-Elementen bestehen und entweder automatisch über die Zeilenerkennung oder manuell über einen gewünschten Bildausschnitt gelegt werden. Es unterstützt keine komplexeren geometrischen Figuren wie Polygone. Seine Stärken kann das System vor allem bei Texten mit klar abgrenzbarer Zeilenstruktur und einem symmetrischen Textlayout zum Ausdruck bringen. Ein sehr interessantes Werkzeug stellt auch das innerhalb des Forschungsverbundes TextGrid entwickelte Text-Imagelinking-Tool dar.⁴³⁶ Dieses besitzt im Gegensatz zu TILE die Möglichkeit, Bildinformationen nicht nur über Rechtecke zu referenzieren, sondern Bildausschnitte auch über beliebige geometrische Figuren in Form von Polygonen zu markieren. Allerdings ist dieses System nicht browserbasiert. Da das angestrebte Ziel dieser Arbeit aber unter anderem darin besteht, ein System zu implementieren, das in kollaborativen browserbasierten Szenarien eingesetzt werden können soll, wurde auf der Basis von SVG und javascript ein eigenes Bildbearbeitungs- und Darstellungstool zur Visualisierung textgenetischer Prozesse entwickelt, das folgende Anforderungen unterstützt⁴³⁷:

1. Das Modul sollte vollständig browserbasiert sein.
2. Es sollte ein lizenzfreies Vektorgraphikformat aus dem Open-Source-Bereich unterstützen.

⁴³⁴ Vgl. auch Eric LECOLINET/Laurent ROBERT/Francois ROLE: Text-image Coupling for Editing Literary Sources, in: Computers and the Humanities 36 2002, S. 49–73.

⁴³⁵ <http://mith.umd.edu/tile/>

⁴³⁶ <http://www.textgrid.de/>

⁴³⁷ Zu den allgemeinen Vorteilen vektorbasierter Grafiken gegenüber Rastergrafiken vgl. z.B. Jeremy J. WISCHUSEN: Scalable Vector Graphics in HTML5. Basic concepts and usage, in: IBM developerWorks 2012, S. 1–22, URL: public.dhe.ibm.com/software/dw/web/wa-scalable/wa-scalable-pdf.pdf. Für den Bereich SVG-basierter Ansätze siehe Hugh A. CAYLESS: Linking Text and Image with SVG, in: Kodikologie und Paläographie im digitalen Zeitalter - Codicology and Palaeography in the Digital Age. Schriften des Instituts für Dokumentologie und Editorik 2 (2009), S. 145–158, URL: <http://kups.ub.uni-koeln.de/2967/>.

3. Es sollte über alle Grundfunktionalitäten des Zoomens, Verschiebens und Rotierens von Images verfügen.
4. Es sollte über unterschiedliche Filterfunktionen verfügen (Kontrast, Helligkeit, Farbwertkorrektur, Negativ-Darstellung).
5. Bildausschnitte sollten mit Hilfe unterschiedlicher geometrischer Figuren in Form von Polygonen mit den Textinformationen verknüpft werden können.
6. Es sollte Überblendungstechniken von Text ermöglichen, die gleichzeitig als Grundlage diplomatischer Darstellungen fungieren können.
7. Es sollte über eine automatische Line- und Word-Recognition-Funktion verfügen, um das exakte Positionieren des Textes auf dem Digitalisat zu ermöglichen.
8. Die Textsegmente sollten durch geeignete Werkzeuge über flexible Formatierungs- (Schriftart, -größe, -Zeilenneigung, Zeichen- und Wortabstände, etc.) und Positionierungsmöglichkeiten verfügen, um den Text bei einer schwierigen Überlieferungssituation, die eine automatische Textpositionierung nicht möglich macht, manuell einfügen und bearbeiten zu können.
9. Die Textsegmente sollten stufenweise einblendbar sein, um die Textgenese visualisieren zu können.

3.3.1 Browserbasierte Vektorgraphikformate - Evaluation

Aus den zahlreichen Entwicklungsbemühungen im Bereich der Vektorgraphikformate seit Beginn der 90er Jahre⁴³⁸ sind inzwischen zwei marktreife Standards hervorgegangen. Die beiden Hauptvertretern von Vektorgraphikformaten im WWW sind das von Macromedia vertriebene FLASH⁴³⁹ und das von einer Arbeitsgruppe des W3C entwickelte und anfänglich vor allem von der Firma Adobe geförderte SVG (Scalable Vector Graphics).⁴⁴⁰ Bei FLASH handelt es sich um ein proprietäres, binäres Vektorformat. Zur Darstellung von FLASH-Grafiken wird ein Browser Plug-in benötigt, das von Macromedia entwickelt wurde und inzwischen standardmäßig ohne Zusatzinstallation zum Auslieferungsumfang der gängigsten Browser gehört. SVG ist neben FLASH das am häufigsten verwendete Vektorgraphikformat in www. Im Gegensatz zu dem proprietären Format von FLASH bietet es zahlreiche Vorteile. SVG ist ein lizenzfrei erhältliches Format, das vollständig in XML implementiert ist. Es wird vom W3C standardisiert und bietet die Möglichkeit, in Browserumgebungen via javascript manipuliert zu werden. Im Gegensatz zu FLASH benötigt SVG kein Browser Plug-in. SVG-Dokumente sind von Suchmaschinen indexierbar und unterstützen das DOM Level 1 und 2. Die anfänglichen Schwierigkeiten, sich gegenüber dem FLASH-Format zu

⁴³⁸ Vgl. Till VOSWINCKEL: XML - SVG Presenter Strukturierte Multimedia-Präsentation im Web, Fachhochschule Furtwangen 2003, URL: www.carto.net/papers/till_voswinckel/paper_till_voswinckel_xml_svg.pdf.

⁴³⁹ <http://www.adobe.com/de/products/flash.html>

⁴⁴⁰ <http://www.w3.org/Graphics/SVG/>

behaupten, können inzwischen als überwunden gelten, da SVG als offizieller Bestandteil von HTML5 nativ von den meisten modernen Browsern unterstützt wird. Die Tabelle 3.1 fasst Vor- und Nachteile von SVG und FLASH zusammen.⁴⁴¹ Letztendlich ausschlaggebend für die Entscheidung, SVG zu verwenden, waren die zahlreichen Vorteile, die dieser W3C-Standard bietet.

3.3.2 Grundarchitektur

Die Applikation verfügt über ein kleines Navigations- und Bearbeitungspanel am linken oberen Bildrand der Image-Ansicht (in Abbildung 3.16 rot markiert).

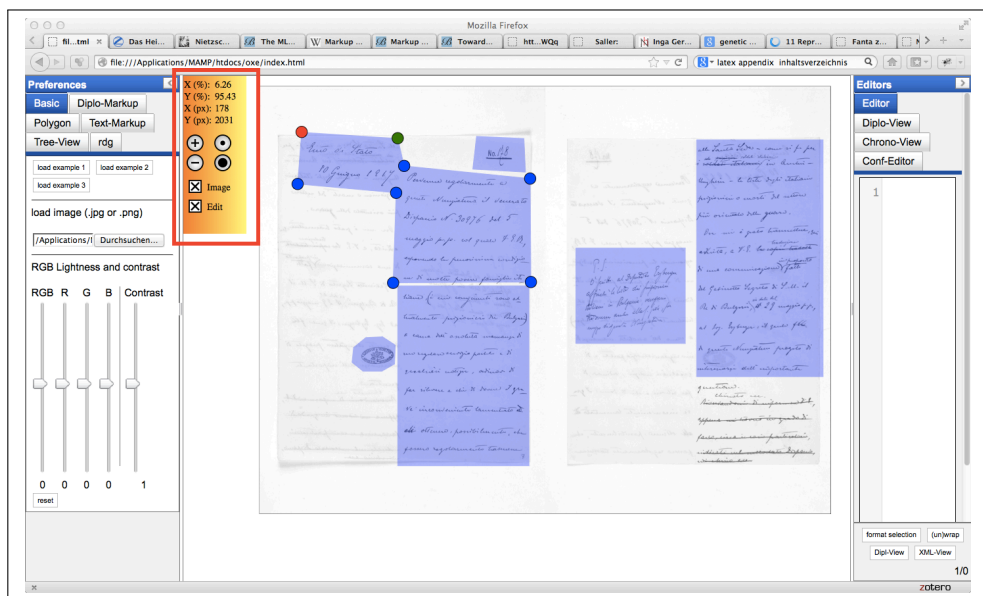


Abbildung 3.16 SVG-View im Polygonbearbeitungsmodus

Hier werden die Bildkoordinaten in Pixeln und Prozent angezeigt und Schaltflächen für die Grundfunktionen des Verkleinerns und Vergrößerns, der bildschirmfüllenden Anzeige und der Originalgröße zur Verfügung gestellt. An eine andere Position verschieben lässt sich das Image durch die üblichen Drag- und Drop-Funktionen. Mit Hilfe eines Doppelklicks auf eine beliebige Stelle im Image, wird dieses automatisch mit den angeklickten Koordinaten als neuem Mittelpunkt zentriert. Über die Bildbearbeitungsfunktionen im Basic-Menü können die verschiedenen Filterfunktionen der SVG-Spezifikation genutzt werden.⁴⁴² Diese werden aktuell nicht von allen Browserherstellern in gleichem Umfang unterstützt, was zum Teil daran liegt, dass SVG erst seit HTML5 eine Schlüsselrolle in der Darstellung vektorbasierter Grafikforma-

⁴⁴¹ Vgl. auch http://www.itk.ilstu.edu/faculty/javila/svg/SVG_overview/svg-vs-flash.htm

⁴⁴² <http://www.w3.org/TR/SVG/filters.html>

SVG

Vorteile	<ul style="list-style-type: none"> • vollständig offener Standard (W3C) • sehr gut dokumentiert • Open-Source • SVG ist XML-Standard, daher: <ul style="list-style-type: none"> • vom Menschen lesbar • unproblematisches Nacheditieren möglich • breite und gut funktionierende Palette von XML-verarbeitenden Tools und Bibliotheken auf dem Markt • SVG-Datei ist nach Text durchsuchbar • SVG-Dokumente sind von Suchmaschinen indexierbar • Unterstützung von DOM Level 1 und 2, XLink (jeder Bereich der Grafik lässt sich verlinken) und Stylesheets (CSS) • Interaktion durch Skriptsprachen (javascript, ecma)
Nachteile	<ul style="list-style-type: none"> • mangelnder Datenschutz bei SVG möglich: einmal ins Netz gestellt, kann es frei kopiert werden

FLASH

Vorteile	<ul style="list-style-type: none"> • größere Marktanteile als SVG, wird von allen gängigen Browsern unterstützt • besserer Datenschutz, da binäres Datenformat
Nachteile	<ul style="list-style-type: none"> • als proprietäres, binäres Datenformat widerspricht es der Open-Source-Idee und kann unvorhergesehenen Veränderungen unterliegen • nachträgliche Bearbeitung der Daten aufgrund der binären Datenhaltung schwerer als bei SVG • nicht nach Text durchsuchbar • Interaktion nur durch spezielle Macromedia-Skriptsprache möglich (Aktion-Script)

Tabelle 3.1 Vergleich von SVG und FLASH

te einnimmt.⁴⁴³ Deshalb beinhaltet das Filtermenü auch nur die Basisfunktionen für Helligkeit, Kontrast und Farbwertkorrektur.⁴⁴⁴

```

1 <svg id="main" xmlns="http://www.w3.org/2000/svg" x="0" y="0" width="0" height
  ="0" viewBox="0 0 0 0" onmousemove="showCoords(evt);" onmousedown="
  showCoords(evt);">
2   <g id="drag" onmousedown="LayerAction(evt)" onmousemove="LayerAction(evt);"
     onmouseup="LayerAction(evt);" onmouseout="LayerAction(evt)">
3   <rect id="mainRect" x="0" y="0" width="0" height="0" />
4   <g id="polyLayer" visibility="hidden">
5     <rect id="polyLayerRect" x="0" y="0" width="0" height="0" visibility="
      visible" />
6     <g id="textCanvas"></g>
7     <g id="polyCanvas"></g>
8   </g>
9   <g id="imageLayer" visibility="visible">
10    <rect id="imageRect" x="0" y="0" width="0" height="0" />
11    <image id="sourceImage" width="0" height="0" filter="url(#f3)" xlink:href=
      "">
12    </image>
13    <rect id="sourceDoc" x="0" y="0" width="0" height="0" />
14  </g>
15  <g id="innerAnimate" visibility="hidden" />
16  </g>
17  <g id="rootAnimate" visibility="hidden" />
18 </svg>

```

Listing 3.9

Das Kernstück der SVG-Applikation bildet die in Listing 3.9 dargestellte Daten-Struktur. Beim Aufruf eines Projekts wird dieses Grundframework mit den entsprechenden Image-Parametern (Image-URL, Bildmaße, Skalierungsfaktor etc.) initialisiert und die verknüpften Polygon- und SVG-Text-Objekte in die jeweiligen Layerelemente (polyCanvas und textCanvas) geladen. Das in der Hierarchie direkt unterhalb des SVG-root-Elements notierte Element mit der id "drag" ist mit den Mouseevent-handlern onmousedown, onmousemove, onmouseup und onmouseout ausgestattet. In der hier aufgerufenen Methode LayerAction wird über event.target der Urheber der Aktion ermittelt und an die entsprechende Funktion weitergeleitet. Über die id des auslösenden Elements wird festgestellt, um welchen Layermodus es sich handelt. Bei der Identifizierung des Image-Layers wird beim onmousedown-Event z.B. die Funktion grabImage(evt) aufgerufen, die den Prozess einer Imageverschiebung initialisiert,

⁴⁴³ Vgl. auch David DAILEY: An SVG Primer for Today's Browsers, in: W3C Editor's Draft 2010, URL: <http://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html>.

⁴⁴⁴ Hier gilt, es in den nächsten Jahren noch gegenüber FLASH-basierten Anwendungen wie z.B. pixlr aufzuholen (<http://pixlr.com/>).

beim Polygon-Layer delegieren zusätzliche Abfragen die Events an die entsprechenden Funktionen. Handelt es sich um ein `circ`-Element eines Polygoneckpunktes wird `grabPolypoint(evt)` aufgerufen (Verschieben eines Polygonpunktes), bei dem `rect`-Element `polyLayerRect` wird ein neuer Polygonpunkt eingefügt. Die Image- und Polygon-Layer werden in SVG durch zwei `g`-Elemente realisiert. Diese `rectangle`-Elemente dienen einzig und allein dem Zweck, zu ermitteln, welcher Layer bei einem mouse-Event aktiv ist. Sie besitzen deswegen die gleiche Größe wie das `root`-Element, da der gesamte Bereich der `viewBox` ausgefüllt sein muss, um über `event.target` an einer beliebigen Position in der SVG-View an die `id` des Layers gelangen zu können. Eine große Rolle hierbei spielt die Position der beiden `rect`-Elemente im Quelltext. Die Rangfolge in Bezug auf die 'Sichtbarkeit' ist im DOM klar geregelt: Ein Element, das im Quelltext hinter einem vorigen Element notiert ist, hat den Vorrang vor diesem. Beim Aktivieren eines Layers muss also die Anordnung dieser Layer-Elemente jeweils aktualisiert werden. Um zu gewährleisten, dass alle Regionen der SVG-View auf den entsprechenden Layer appliziert werden können, haben die `rect`-Elemente die gleichen Abmaße wie die `viewBox` des `SVG-root`-Elements. Über die Funktion `setPolyLayer(evt)` wird beim Aktivieren des Polygonlayers das `g`-Element `PolyLayer` im DOM-Baum an die letzte Position innerhalb des `drag`-Elements gesetzt. Im Gegensatz zu letztem Listing ist in Listing 3.10 der Bearbeitungsmodus aktiviert (`<g id="polyLayer" visibility="visible">`) und das Element hinter den Image-Layer verschoben worden.

```

1 <g id="drag" onmousedown="LayerAction(evt)" onmousemove="LayerAction(evt);"
  onmouseup="LayerAction(evt);" onmouseout="LayerAction(evt)">
2   <rect id="mainMapRect" x="0" y="0" width="0" height="0" />
3
4   <g id="imageLayer" visibility="visible">
5     <rect id="imageRect" x="0" y="0" width="0" height="0" />
6     ...
7   </g>
8
9   <g id="polyLayer" visibility="visible">
10    <rect id="polyLayerRect" x="0" y="0" width="0" height="0"/>
11    ...
12  </g>
13 </g>

```

Listing 3.10

Somit ist sichergestellt, dass alle mouse-Events das entsprechende Verhalten auslösen, welches in diesem Layer auch erwünscht ist, also z.B. das Einfügen und Bearbeiten von Polygonen. Zusätzlich erhalten die Kindelemente (Polgone, Textobjekte) über das Attribut `visibility` den Status der Sichtbarkeit, um die in diesem Layer vorhandenen Objekte anzuzeigen.⁴⁴⁵ Alle in einem Image annotierten Polygone befinden sich

⁴⁴⁵ Es genügt nicht, nur das Attribut des jeweiligen Layers auf `visible` zu setzen, da dadurch nicht

innerhalb des Elements `<g id="canvas"/>`, das wiederum innerhalb des `PolyLayer`-Elements notiert ist. Die `path`-Elemente in `canvas` werden über `getElementsByTagName("path")` selektiert und ihre `visibility`-Attribute mit Hilfe einer `for`-Schleife gesetzt. Analoges gilt für die SVG-Textobjekte zur Integrierung des Transkriptionstextes in die SVG-View. Diese werden in dem Layer `textCanvas` eingefügt und über Iterationen ein- und ausgeblendet. Der so aktivierte Bearbeitungsmodus wird neben der Anzeige des Häkchens in der Edit-Auswahlbox zusätzlich durch ein leichtes Aufhellen der Bildanzeige und das Einblenden evtl. schon vorhandener Polygone visualisiert.

Im Bearbeitungsmodus der Applikation lassen sich die Transkriptionen mit den digitalen Faksimiles auf zwei unterschiedliche Arten verknüpfen. Zum einen mit einfachen geometrischen Formen (Polygonen) und zum anderen mit dem direkten Integrieren des Transkriptionstextes in das Digitalisat. Erstere einfachere Variante ist dann geeignet, wenn kein Wert darauf gelegt wird, eine genaue diplomatische Transkription zu erstellen, sondern es ausreicht, Verknüpfungen von Textregionen des Digitalisats und denen der Transkription zu erstellen, um sie etwa durch Anklicken einer markierten Textpassage farblich hervorzuheben. Die zweite Variante sollte dann angewendet werden, wenn eine topographisch genaue diplomatische Transkription erstellt werden möchte, oder schwer lesbare Textstellen durch Überblendungstechniken sichtbar gemacht werden sollen. Im folgenden werden Datenstruktur und Funktionsweise des einfacheren Polygonlayers beschrieben.

3.3.2.1 Polygonlayer

Die für diese Anwendung in SVG implementierten Polygone besitzen die Grundstruktur aus Listing 3.11. Im eigentlichen Sinne handelt es sich nicht um SVG-Polygon-Elemente, sondern um eigens definierte SVG-Objekte, die aus einem Containerobjekt bestehen (`g`), in das ein `path`-Element zur Darstellung der äußeren Form und `circle`-Elemente zur Visualisierung der Polygoneckpunkte integriert werden. Jedes `g`-Element ist über ein `id`-Attribut eindeutig referenzierbar. `Path`-Elemente in SVG sind geeignet, komplexe geometrische Objekte darzustellen.⁴⁴⁶ Mit ihrer Hilfe lassen sich alle geometrischen Basisformen realisieren, die in SVG auch eigene Elementdefinitionen besitzen (Rechteck `<rect>`, Kreis `<circle>`, Ellipse `<ellipse>`, Linie `<line>`, offene `<polyline>` oder geschlossene Polygonform `<polygon>`).⁴⁴⁷ Der Vorteil von `path`-Elementen besteht darin, dass sie über das DOM-Interface `SVGPathSeg` verfügen und somit komfortabel auf einzelne Koordinatenpaare zugegriffen werden kann. Dadurch entstehen einige zentrale Vorteile bei der DOM-basierten Verarbeitung der Objekte gegenüber `polyline` und `polygon`. Beim Modifizieren der Polygonobjekte durch Einfügen oder Verschieben eines Punktes z.B. müsste bei `polyline` und `polygon` der durch Leerstellen getrennte Koordinatenstring aus dem Attribut `point` immer wieder manuell geparst, an der entsprechenden Stelle aktualisiert und neu gesetzt werden. Bei der

automatisch alle Kindelemente auch den Status der Sichtbarkeit erhalten. Vielmehr muss über sämtliche Polygonelemente innerhalb des Layers iteriert und auch deren `visibility`-Eigenschaft aktualisiert werden.

⁴⁴⁶ <http://www.w3.org/TR/SVG/paths.html#PathElement>

⁴⁴⁷ <http://www.w3.org/TR/SVG/shapes.html>

Verwendung von path kann über das Interface SVGPathSegList direkt auf die Koordinaten zugegriffen werden. Im Gegensatz zu den eigentlich vom W3C empfohlenen polyline- oder polygon-Elementen ist der Funktionsumfang von path weitaus größer. Genau wie bei polyline und polygon werden im path-Element die Punktkoordinaten in einem Attribut gespeichert. In diesem d-Attribut können neben den Koordinaten auch spezifische Verarbeitungsanweisungen festgehalten werden. So lassen sich etwa über das zusätzliche Einfügen von Parametern in das d-Attribut horizontale (H) und vertikale Linien (V), elliptische Bögen (A - Elliptical Arc) oder unterschiedliche Arten von Bézier-Kurven (C - CurveTo, S - Shorthand/Smooth CurveTo, Q - Quadratic Bézier CurveTo)⁴⁴⁸ verwirklichen.

```

1 <g id="polyCan_3" visibility="visible">
2   <path d=" M1.98164e+07,1.05985e+07 L1.92216e+07,1.65126e+07 L2.52679e+
3         07,1.5984e+07 L2.60939e+07,1.06315e+07Z"
4         onclick="activatePoly(evt)"
5         onmouseover="setHighlightPoly(evt)"
6         onmouseout="removeHighlightPoly(evt)"
7         visibility="visible"
8         id="path_3"
9         style="fill:blue;opacity:0.3;" />
10  <circle id="3_0" type="circ" cx=" 19816350" cy=" 10598490" r="200000"
11        style=" fill:red;stroke:black;" visibility="visible"/>
12  <circle id="3_1" type="circ" cx="19221634" cy="16512601" r="200000"
13        style="fill:blue;stroke:black;" visibility="visible"/>
14  <circle id="3_2" type="circ" cx="25267905" cy="15983966" r="200000"
15        style="fill:blue;stroke:black;" visibility="visible"/>
16  <circle id="3_3" type="circ" cx="26093898" cy="10631529" r="200000"
17        style=" fill:green;stroke:black;" visibility="visible"/>
18 </g>

```

Listing 3.11

Im vorliegenden Beispiel startet das Polygon bei Punkt M1.98164e+07,1.05985e+07. M (MoveTo) signalisiert, dass es sich hier um den Startpunkt des Pfades handelt. Das zweite Punktkoordinatenpaar L1.92216e+07,1.65126e+07 beginnt mit einem L (LineTo). Hierdurch wird vermerkt, dass eine Linie vom vorherigen Punkt zu diesem Punkt gezeichnet wird. Dies wiederholt sich bei den beiden folgenden Punkten. Schließlich endet der Attributwert mit dem Parameter Z (ClosePath), der angibt, dass es sich um ein geschlossenes Polygon handelt, also vom letzten Punkt eine Linie zum Startpunkt gezeichnet wird. Das style-Attribut fill:blue;opacity:0.3 von path bewirkt das Rendering des Polygons mit der Farbe Blau und einem Transparenzwert von 0.3 (prozentuale Durchlässigkeit des Hintergrundes). Desweiteren verfügt path über ein onclick-Event, dass die Methode activatePoly(evt) aufruft und den Aktivierungsprozess des Poly-

⁴⁴⁸ <http://www.w3.org/TR/SVG/paths.html#PathDataCubicBezierCommands>

gons auslöst. Die onmouseover- und onmouseout-Events bewirken beim Überfahren des Polygons mit dem Mauszeiger eine Farbveränderung. Die circle-Elemente, die als Ankerpunkte zum Modifizieren der Polygonform dienen, werden auf der gleichen Hierarchieebene wie path notiert und besitzen jeweils die gleichen Koordinaten, wie ihr jeweiliges Pendant im Pfad-Attribut d. Die Koordinaten $cx=19816350$, $cy=10598490$ des ersten circle-Elements z.B. entsprechen (an der 5ten Nachkommastelle gerundet) dem Startpunkt des Polygons $d=M1.98164e+07,1.05985e+07$.

Bei aktiviertem Edit-Layer stehen alle Funktionen zur Verfügung, um beliebige Bildbereiche mit SVG-Polygonen auszuzeichnen. Ein einfacher Mausklick bewirkt das Einfügen des ersten Punktes eines neuen Polygons. Jeder weitere Mausklick in diesen SVG-Layer hat das Eingruppieren eines weiteren Punktes zwischen dem zuletzt eingefügten und dem ersten Polygonpunkt zur Folge. Der erste Polygonpunkt erscheint dabei rot, der letzte grün, die übrigen blau. Beim Einfügen eines neuen Punktes wird der Farbwert des aktuellen circ-Elements auf grün gesetzt und der des zuvor eingefügten auf blau. Jedes Polygon wird schon beim Zeichnen mit einem transparenten Blauton gefüllt. So lässt sich der darunter liegende Bildausschnitt noch gut erkennen und gleichzeitig die Umrisse des noch nicht fertigen Polygons anzeigen. Polygone müssen nicht extra geschlossen werden. Um ein neues Polygon anzulegen, genügt es, auf die halbtransparente Fläche des path-Elements zu klicken. Dadurch werden die Eckpunkte (circles) ausgeblendet und somit signalisiert, dass der Layer bereit ist zum Einfügen eines neuen Polygons. Bei erneutem Mausklick auf eine freie Fläche des Images wird mit einem neuen Polygon begonnen. Ein bereits geschlossenes Polygon kann zurück in den Bearbeitungsmodus versetzt werden, indem man es durch einen Mausklick in die gefüllte Polygonfläche wieder aktiviert. Bei aktiven Polygonen werden die Eckpunkte angezeigt, bei inaktiven nur die transparenten farblich gefüllten Flächen. So lässt sich schnell erkennen, welches Polygon sich gerade im Bearbeitungsmodus befindet. Die Eckpunkte eines Polygons können verschoben werden, indem man einen Mausklick auf einen Polygonpunkt ausführt und diesen mit gedrückter Maustaste an eine andere Position verschiebt. Beim Loslassen der Maustaste verbleibt der Polygonpunkt an dieser Stelle. Das Verschieben der Punkte wird mittels javascript durch das mousemove-Event überwacht. Bei jedem drag-Vorgang bei gedrückter Maustaste werden fortlaufend die X und Y Koordinaten der Mausposition überwacht und mittels dieser das transform-Attribut von SVG genutzt, um die entsprechenden SVG-Elemente zu verschieben.

Da es sich bei einem Polygon um ein Objekt bestehend aus circle- und path-Elementen handelt, wird nicht nur das transform-Attribut des jeweiligen circles aktualisiert, sondern auch das zugehörige path-Element. In der pathSegList,⁴⁴⁹ über die jedes path-Objekt verfügt, sind die einzelnen Pfadpunkte gespeichert. Beim Einfügen eines circles erhält dieser eine feste id, die mit der Listenposition der pathSegList übereinstimmt (`<circ id="circ_1_3".../>` entspricht dem dritten Polygonpunkt im ersten Polygon). Die Koordinaten des verschobenen dritten circle Elements werden nun genutzt um auch pathSegList an der dritten Position zu aktualisieren. Für den Betrachter wirkt es damit so, als seien circle und path eine Einheit, von der Da-

⁴⁴⁹ <http://www.w3.org/TR/SVG/paths.html#InterfaceSVGPathSegList>

tenstruktur her sind es jedoch separate Elemente. Punkte können Polygonen nicht nur am Ende hinzugefügt werden (zwischen rotem Anfangspunkt und grünem Endpunkt), sondern auch zwischen zwei beliebigen Punkten. Befindet sich der Mauszeiger auf einem circ-Element, wird automatisch ein kleiner gelber Punkt auf halber Distanz zwischen Mauszeigerpunkt und vorherigem eingeblendet (Abbildung 3.17, links). Bei einem Doppelklick auf den Mauszeigerpunkt wird an der Position des gelben Punktes ein circle-Element im Polygonobjekt eingefügt und die pathSegList aktualisiert.

Sind nicht Polygone, sondern Rechtecke erwünscht, fügt man nur den ersten Polygonpunkt ein und zieht per drag and drop diesen Punkt an eine beliebige Stelle im Image (Abbildung 3.17, rechts). Dabei wird automatisch ein

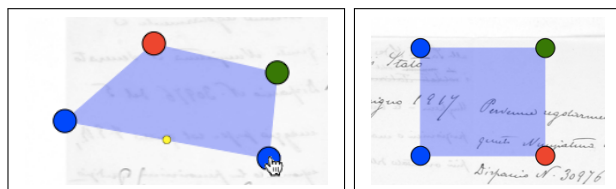


Abbildung 3.17

Rechteck aufgespannt. Dieses kann durch Ziehen beliebiger Eckpunkte auf die richtige Größe gebracht und an die richtige Position gesetzt werden. Integriert man einen weiteren circle-Punkt in das rechteckige Polygon, so verliert es dabei seine soeben beschriebenen rechteckigen Eigenschaften und verhält sich wieder wie ein normales Polygon. Dieser Prozess ist nicht wieder rückgängig zu machen.

Eine Verknüpfung mit einem Textsegment wird angelegt, indem das Polygon-Objekt und das zu verknüpfende Text-Widget aktiviert werden (Mausklick auf beide Elemente) und anschließend über die Schaltfläche 'link to text' im Menü "Polygon" die id des Polygonobjektes in das jQuery-Data-Objekt des Markup-Widgets eingefügt wird.

3.3.3 Wie kommt der Text ins Digitalisat?

Die zweite Variante der Verknüpfung von Text und Image besteht in der Möglichkeit, den Text direkt ins Digitalisat zu integrieren. Hier existieren zwei Ansätze, um dieses Ziel zu erreichen. Zum einen ein manueller Modus und zum anderen ein automatisierter Prozess, der über Zeilen- und Wortsegmentierungsalgorithmen die Textpositionen ermittelt.

3.3.3.1 Manuelles Einfügen

Beim manuellen Einfügen werden die Polygoneigenschaften des in Abschnitt 3.3.2.1 beschriebenen Polygonlayers genutzt. Um einen markierten Text im Editor in das Digitalisat einzufügen, wird ein Polygon benötigt, das aus lediglich zwei Punkten besteht (im Prinzip also nur eine Linie, von der SVG-Struktur her aber ein Polygonobjekt). Ist dieses Polygon in das Digitalisat eingezeichnet worden und befindet sich gleichzeitig ein Markup-Widget in aktivem Zustand, kann über die Schaltfläche setDiplo im Navigationsmenü Diplo-Markup das Textsegment eingefügt werden (Abbildung 3.18). Der einzufügende Text wird über startLine, endLine, startChar und endChar

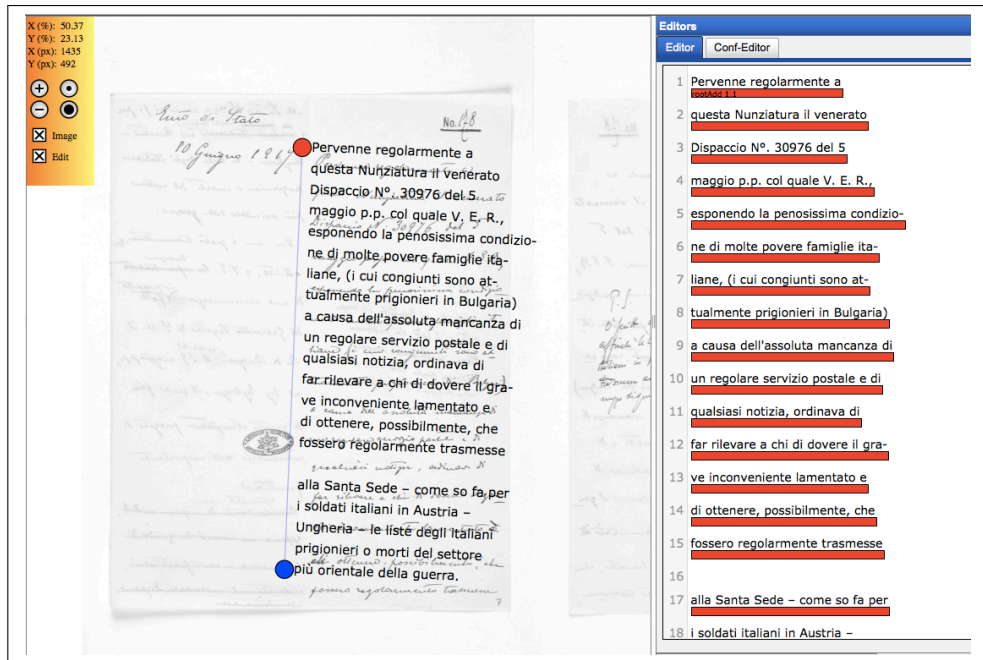


Abbildung 3.18

der markierten Linewidgets aus der Editorinstanz ermittelt und in ein SVG-Fragment transformiert (Listing 3.12).

```

1 <g id="PolyText_1" inclination="rotate" x="725" y="421" transform="rotate
  (0.38952031201592197, 695, 411)">
2   <text x="725" y="421" dxVal="0" wSpace="0" font-size="40" font-family="
    Verdana" ln="root_ln_0">Pervenire regolarmente a </text>
3   <text x="725" y="536.5740993864092" dxVal="0" wSpace="0" font-size="40"
    font-family="Verdana" ln="root_ln_1">questa Nunziatura il venerato </
    text>
4   ...
5 </g>

```

Listing 3.12

Für jedes Linewidget wird dabei ein text-Element erzeugt. Ausgerichtet werden die Zeilenobjekte an den beiden Punkten des Polygons. Der Text des ersten Linewidgets erhält dabei die x-Koordinate des roten Startpunktes und die um den Radius des circle-Elements verschobene y-Koordinate (damit der Text nicht direkt auf dem Polygonpunkt beginnt). Die Positionierung der restlichen Zeilen wird aus dem Abstand

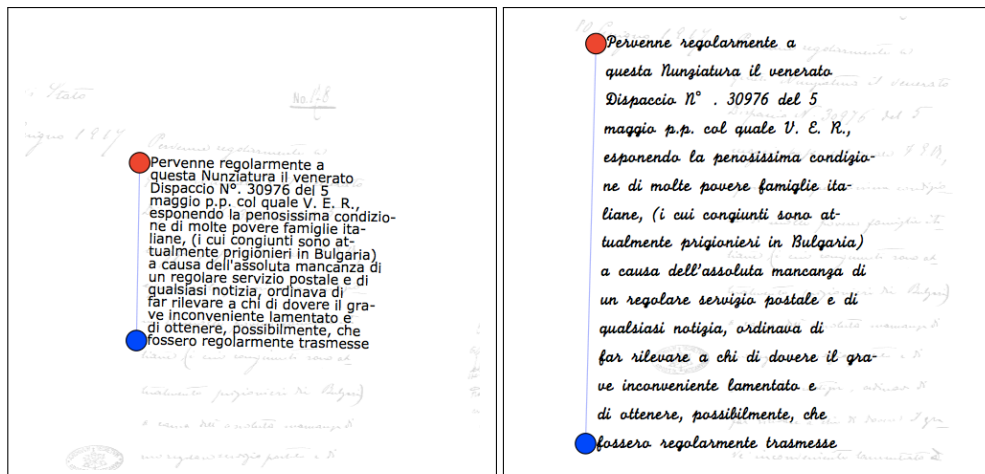


Abbildung 3.19 unterschiedlicher Zeilenabstand und Fonttype

der x- und y-Koordinaten der Punkte errechnet und gleichmäßig verteilt, bis die letzte Zeile am Endpunkt eingefügt worden ist.

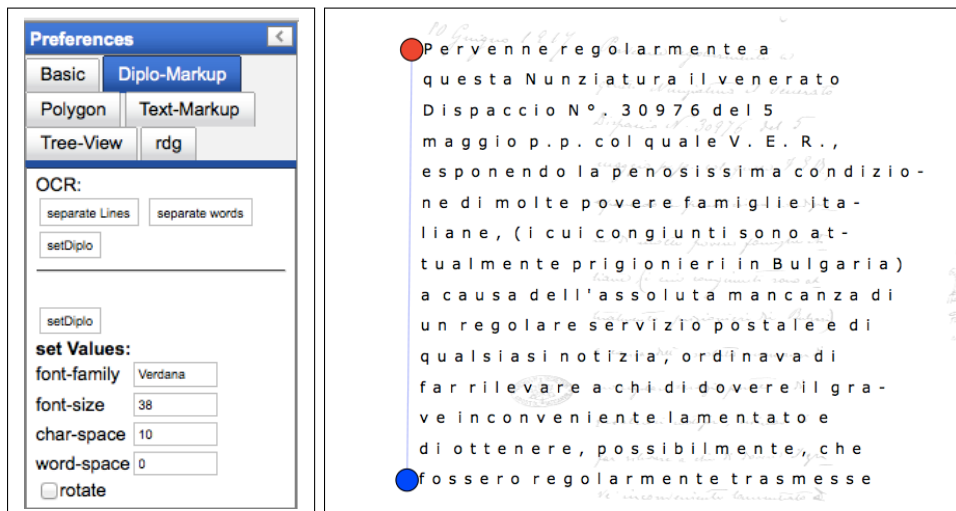


Abbildung 3.20 Charspace

Diese grobe Ausrichtung lässt sich weiter optimieren, indem die beiden Polygonpunkte an die gewünschte Stelle verschoben werden. Die Positionierung der Zeilen wird dabei jeweils neu berechnet. Auf diese Art und Weise lässt sich auch leicht der Zeilenabstand variieren (Abbildung 3.19). Über das Formatierungsmenü im Diplo-Markup-Panel (Abbildung 3.20 links) können weitere Einstellungen vorgenommen werden. Da die Unterstützung von SVG, wie schon erwähnt, in den aktuellen Browsern noch nicht

zu 100% identisch ist, kann der Zeichenabstand noch nicht browserübergreifend mit dem dafür vorgesehenen SVG-Attribut 'kerning' implementiert werden.⁴⁵⁰ Um trotzdem diese Funktionalität in allen Browsern zu gewährleisten, kann statt dessen der Abstand zeichenweise über das dx-Attribut gesetzt werden. Dazu muss die Länge der Zeile ermittelt und in einer Schleife über die einzelnen Zeichen der Abstand definiert werden:

```
1 <text dx="0 10 10 10 10 10 10 0 10 10 10 10 10 10 10 0 10 10 0 10 10 10
    10 10 10 10 10">questa Nunziatura il venerato</text>
```

Listing 3.13

Dieser Wert kann eine sinnvolle Verkürzung (negative Werte sind auch gültig) oder Verlängerung der Zeilenlänge bewirken, um die Textelemente möglichst deckungsgleich auf dem Digitalisat platzieren zu können (Abbildung 3.20). Mit dieser Methode

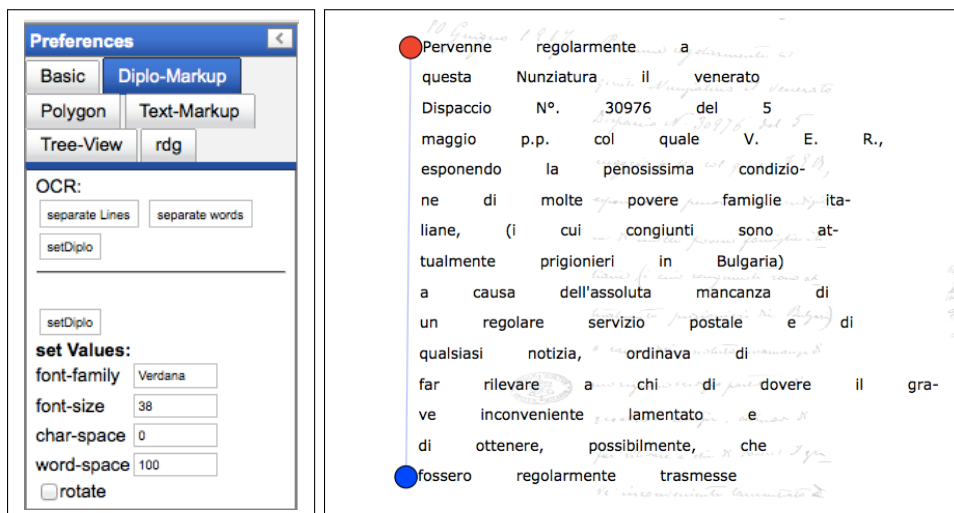


Abbildung 3.21 Wordpace

kann auch auf einfache Weise eine Wortabstandseigenschaft implementiert werden (Abbildung 3.21). Hierbei werden einfach alle Leerzeichen der Zeile durch einen definierten Wordspacewert ersetzt:

```
<text dx="0 0 0 0 0 100 0 0 0 0 0 0 0 0 100 0 0 100 0 0 0 0 0 0 0">
    q u e s t a   N u n z i a t u r a   i l   v e n e r a t o
```

⁴⁵⁰ Von Safari z.B. wird dieses Attribut unterstützt, von Firefox nicht

Dies kann z.B. dann sinnvoll sein, wenn die Wortabstände einer Handschrift nicht denen des verwendeten Schrifttyps entsprechen.

Die so gewonnenen Informationen ließen sich auch dazu verwenden, Regelsets für typische Zeichen- und Wortabstände von Schreibern zu erstellen mit komplexeren Kriterien, wie z.B. der Unterscheidung von Zeichenlänge bei Groß- und Kleinschreibung, Wortanfängen und -enden, numerischen Zeichen, Interpunktionen oder Kombinationen bestimmter aufeinander folgender Zeichen etc. Diese Handschriftencharakteristika könnten global gespeichert werden, und jeder Bearbeiter hätte so die Möglichkeit, bei Identifizierung eines bestimmten Schreibers, die jeweiligen Parameter zu laden und die Abstände automatisch zu generieren. In einem standardisierten Format könnten diese auch frei verfügbar gemacht und in einer zentralen Handschriftendatenbank verwaltet werden.⁴⁵¹ Der Aufwand für so eine Vorabanalyse wäre allerdings nur für eine größere Menge von Textzeugen eines Autors zu rechtfertigen. Darüberhinaus könnten solche Analysen auch dazu dienen, die in Abschnitt 1.2.4.2 schon erwähnten Featuretabellen zur automatischen Texterkennung zu verbessern. Auch wenn die Zeichenabstände

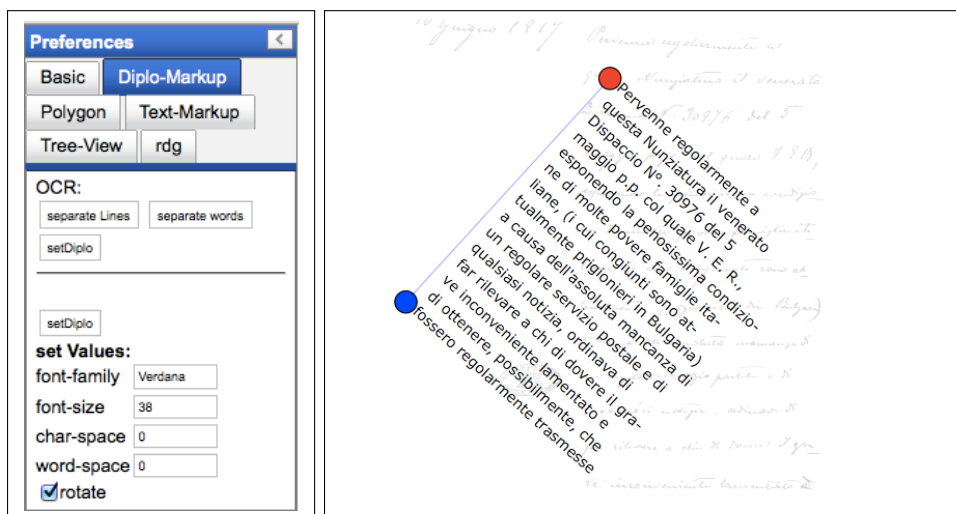


Abbildung 3.22 Mit Rotation

innerhalb einer Handschrift nie ganz identisch sein werden, so ließen sich doch statistische Näherungswerte daraus gewinnen und diese z.B. bei der Zeichensegmentierung einsetzen. Die Werte könnten wertvolle Hinweise darüber liefern, wie groß die Wahrscheinlichkeit ist, dass es sich bei einem automatisch segmentierten Textfragment um ein bestimmtes Zeichen/Wort handelt oder ob das Segment überhaupt ein Textzeichen ist. Je genauer die Zeichenabstände und -längen eines erkannten Textfragments mit den aus den Featuresets gewonnenen Durchschnittswerten übereinstimmen, desto

⁴⁵¹ Eventuell wäre hier zu bedenken, dass sich die Charakteristika einer Handschrift über die Jahre verändern können. Es wäre also auch zu überlegen, die Handschriften-Featuresets in unterschiedliche Produktionsphasen zu untergliedern.

größer wäre die Wahrscheinlichkeit, dass die Texterkennung ein richtiges Ergebnis geliefert hat. Mit dem Auswahlfeld Rotation lässt sich die Schriftneigung anpassen. Ist das Feld aktiviert und verschiebt man einen Punkt des Polygons, so bildet der andere Punkt den Rotationsmittelpunkt. Bei jeder neuen Mauszeigerbewegung wird dabei der Rotationswinkel neu berechnet (siehe Abbildung 3.22). Wird die Rotation deaktiviert, rotieren die Zeilen nicht mehr, richten sich aber dennoch durch horizontale Verschiebung auf der y-Achse aus. Damit lassen sich die für manchen Schreibertypus charakteristischen Abweichungen der Zeilenanfänge simulieren (Abbildung 3.23).

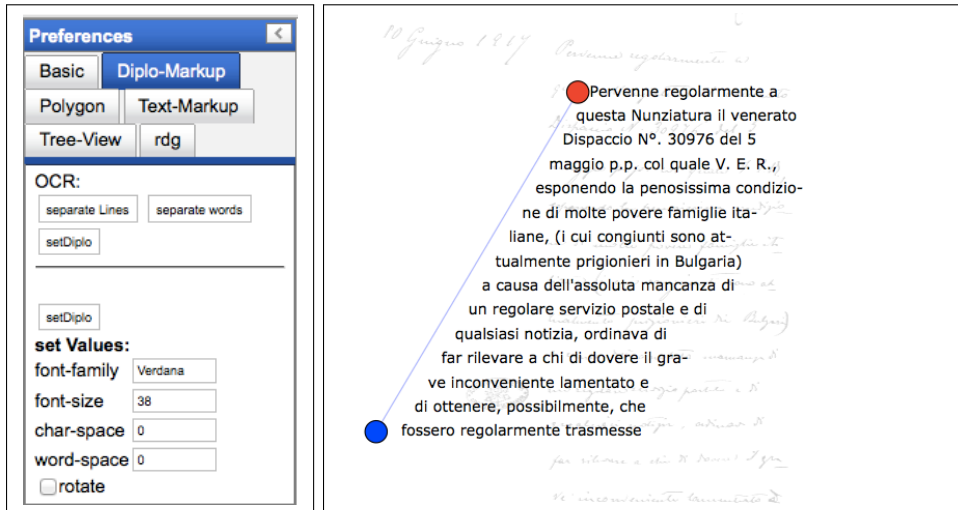


Abbildung 3.23 Ohne Rotation

Für unser Beispiel von vorhin (Abbildung 3.18) kann somit eine schon recht genaue Positionierung des Textes erreicht werden (Abbildung 3.24). Hier ist der Text nicht zeilengenau positioniert worden, sondern aufgrund der günstigen Gegebenheiten (großer Zeilenabstand) jeweils unter dem Text, auf ein fortlaufendes Ein- und Ausblenden des Textes oder des Digitalisats kann hier also im Prinzip verzichtet werden. Die SVG-Text-Objekte und ihre Formatierungseigenschaften werden in der in Abschnitt 3.2.12 beschriebenen stand-off-Datenstruktur gespeichert und jeweils dynamisch zur Laufzeit (also erst beim Einblenden der Textelemente in der SVG-View) neu berechnet. Die im letzten Kapitel beschriebenen manuellen Werkzeuge und Konfigurationsmöglichkeiten können den Bearbeiter beim Einfügen des Transkriptionstextes besonders in solchen Situationen unterstützen, die ein automatisches Positionieren des Textes nicht zulassen. Die Gründe dafür können z.B. ein sehr komplexes Layout, schwer erkennbare Zeilenstrukturen oder Beschädigungen des Dokumentes sein. In allen anderen Fällen kann es dagegen sinnvoll sein, die Zeilen- und evtl. auch Wortsegmentierungen durch einen maschinellen Prozess zu unterstützen und somit das Integrieren des Textes zu automatisieren. Wie schon in Kapitel 1.2.4.2 beschrieben, existieren eine Reihe unterschiedlicher Lösungsansätze und Algorithmen zu diesem Thema. Im folgenden soll ein relativ einfacher Algorithmus verwendet werden.

Die Arbeit verfolgt nicht das Ziel, auf diesem Gebiet neue Methoden zu entwickeln. Vielmehr soll hier demonstriert werden, dass sich solche Algorithmen auch als Werkzeuge in einem browserbasierten Umfeld einsetzen lassen und vor allem, dass dieses Problem auch clientseitig gelöst werden kann. Der Vorteil besteht darin, dass ein Image so nicht erst zu einer Serverapplikation übertragen werden muss und die oft zeitintensiven Berechnungen dort ausgeführt und dass Ergebnisse anschließend wieder zurück gesendet werden. Dies verlagert die Rechenlast zwar auf die Seite des Clients, was aber durchaus auch einige Vorteile bietet. Es besteht z.B. nicht die Gefahr, dass bei vielen gleichzeitigen Serveranfragen unnötige Wartezeiten entstehen. Der lokale Datentransfer ist wesentlich schneller und der Algorithmus kann auch bei nicht vorhandener Konnektivität ausgeführt werden. Was serverseitig zur Verfügung gestellt werden könnte, wäre eine Bibliothek unterschiedlicher Algorithmen, die für spezifische Anwendungsfälle dynamisch nachgeladen würden.⁴⁵² Dabei könnten die browserseitigen Line- und Wordrecognition-Funktionen von den neu geladenen überschrieben werden. Hier sollte dann für einen genormten In- und Output dieser Funktionen Sorge getragen werden. In der Regel würden dazu beim Input die Abmaße (in Pixel oder prozentualen Koordinaten) des zu verarbeitenden Textausschnitts (evtl. auch in nicht rechteckiger, sondern polygonaler Form) benötigt werden und beim Output Objekte, die die Segmentierungskordinaten in standardisierter Form enthalten (z.B. XML oder SVG). Wie dies in SVG aussehen könnte wird im folgenden beschrieben.

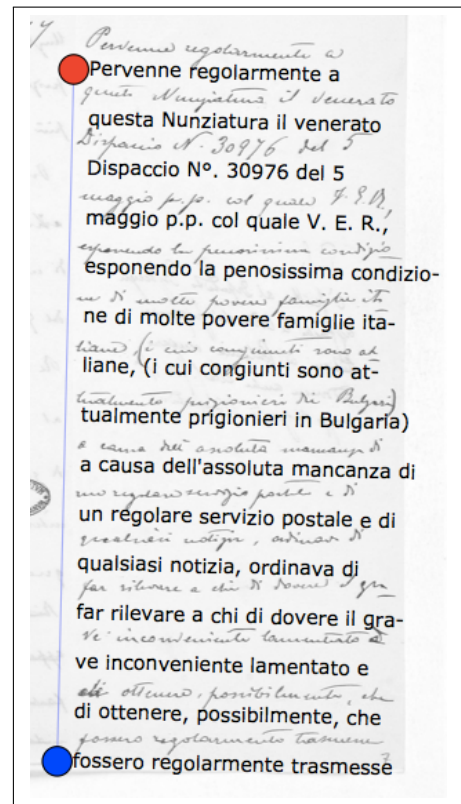


Abbildung 3.24

⁴⁵² Hierzu bemerkt schon Fenton, dass nahezu jedes Projekt andere Anforderungen an die Methoden bzw. Algorithmen stellt. Eine Sammlung von flexibel erweiterbaren Algorithmen wäre hier sicherlich von großem Nutzen. Vgl. Eileen Gifford FENTON/Hoyt N DUGGAN: Effective Methods of Producing Machine-Readable Text from Manuscript and Print Sources, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD/Katherine O'BRIEN O'KEEFE/John UNSWORTH, S. 241–254, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/duggan.xml.

3.3.3.2 Automatisches Einfügen

Der unter dem Namen HTML5 firmierende Webstandard, der sich gerade in der Browserwelt durchzusetzen beginnt, bietet interessante und vielfältige Möglichkeiten, eine javascriptbasierte clientseitige Implementation dieser Art von Segmentierungsalgorithmen umzusetzen. Das in diesem Standard neu eingeführte canvas-Objekt ermöglicht über die Funktion `canvas.getContext('2d').getImageData(x, y, width, height)` den pixelgenauen Zugriff auf Rastergrafiken. Diese Funktionalität macht sich die hier beschriebene Applikation zunutze. Vom geladenen Image (jpeg oder png) muss zuerst über `canvas.getContext('2d').drawImage(img, x, y, width, height)` eine DOM-Instanz des rechteckigen zu segmentierenden Bildausschnitts angelegt werden. Über `canvas.getContext('2d').getImageData` gelangt man so an RGBA-Informationen eines jeden Pixels. Die Koordinaten für das Rechteck werden gewonnen, indem man ein rechtwinkliges SVG-Polygon (wie unter Abschnitt 3.3.2.1 beschrieben) im Bearbeitungslayer der SVG-Anwendung aufspannt. Aus den vier circle-Koordinaten werden `x`, `y`, `width` und `height` berechnet. Über das `image.data`-Array kann jetzt zeilenweise

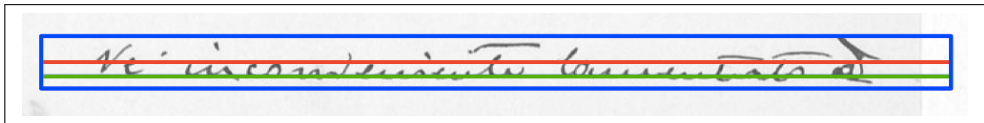


Abbildung 3.25 'Falsche' Zeilenmitte rot (Hälfte der Boundingbox-Höhe), 'richtige' grün (Maximum der addierten Alphakanalwerte einer Pixelzeile)

(`image.width`) iteriert und der Alphakanalwert ermittelt werden. Der Beispielalgorithmus addiert die Alphawerte jeder Zeile und entscheidet aufgrund eines Schwellwerts, ob es sich um eine `whiteLine` handelt (Zeilenzwischenraum) oder ob sie Bestandteil einer handschriftlichen Zeile ist. So lassen sich die Boundingboxen der Zeilen ermitteln und die SVG-Text-Objekte oberhalb, unterhalb oder mittig platzieren.

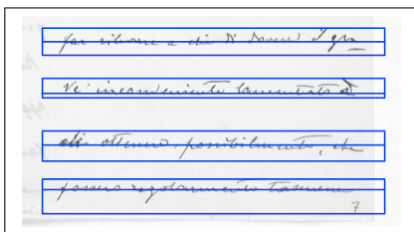


Abbildung 3.26

Zusätzlich erhält man über den maximalen Schwellwert des Alphakanals einer jeden so erkannten Zeile die `centerLine`. Diese kann z.B. genutzt werden, um eine exakte Positionierung des Textes auf der Zeile zu erreichen, die bei großen Ober- oder Unterlängen mit einer einfachen Halbierung der boundingbox-Höhe zu ungenau wäre (Abbildung 3.25). Für Zeilenverläufe, deren Unter- oder Oberlängen ineinanderfließen, ist es bei

dem hier implementierten Algorithmus nicht möglich, die Boundingboxen zu ermitteln. Hier muss man sich mit dem Maximalwert der `centerLine` begnügen. Die mit dem Linerecognition-Algorithmus erzeugten Boundingboxen inklusive `centerLines` für das Beispiel aus vorigem Kapitel (Abbildung 3.24) sind in Abbildung 3.26 zu sehen. Im Gegensatz zum manuellen Verfahren, das einen dynamischen, aber immer

gleichmäßigen Zeilenabstand generiert, werden mit dieser Methode die unterschiedlichen Zeilenabstände automatisch erkannt und die Ankerpunkte zum Einfügen des Textes auch bei variierenden Zeilenabständen richtig gesetzt.

Für die Überblendungstechniken lassen sich die Zeilen wahlweise oberhalb der Boundingbox, unterhalb, oder für die diplomatisch genaueste Transkription auf der centerLine positionieren. Da die Methode `getImageData` nur rechteckige Bildausschnitte liefert, kann es zu Schwierigkeiten beim Identifizieren der relevanten Zeilensegmente kommen. Eine Lösung könnte darin bestehen, das eingefügte nicht rechteckige Polygon mit einer Boundingbox zu umgeben, die sich aus den maximalen und minimalen x- und y-Koordinaten des Polygons errechnen ließe und anschließend die Alphakanalwerte der außerhalb des Polygons liegenden Pixel vor dem Verarbeiten auf 0 zu setzen (Abbildung 3.27) oder umgekehrt alle zu exkludierenden Textsegmente mit einem Polygon zu versehen und dort die Werte zu nullen. Um eine exaktere deckungsgleiche Überblendung zu erlangen, könnte eine Segmentierung der einzelnen Wörter hilfreich sein. Eine erste Betaversion steht hierfür im Diplo-Markup-Menü zur Verfügung. Mit dem Button 'separate words' können die zuvor segmentierten Zeilen in Wörter zerlegt werden.

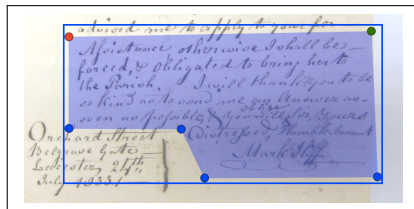


Abbildung 3.27



Abbildung 3.28

Die Funktionsweise des Algorithmus ist im Prinzip die selbe, die auch schon bei der Zeilensegmentierung zum Einsatz kam, nur dass die Boundingboxes der Zeilen nun in horizontaler Richtung durchlaufen und so die vertikalen `whiteLines` ermittelt werden. Hierzu existiert wieder ein Schwellwert, mit dem diesmal angegeben wird, wie viele `whiteLines` ein Leerzeichen zwischen einem Wort markieren. Das Ergebnis ist in Abbildung 3.28 zu sehen. Hiermit können nun die exakten Koordinaten der Zeilenanfänge ermittelt werden (der variierende Zeilenbeginn ist deutlich zu

erkennen).⁴⁵³ Im Prinzip ist es nun möglich, die `centerLines` wieder für jedes segmentierte Wort neu zu setzen und damit jedes einzelne Wort auszurichten. Es ist dabei allerdings zu bedenken, dass eine solche Ausrichtung zwar exaktere Überblendungen ermöglicht, aber auch bei starken Abweichungen den Lesefluss behindern könnte. Es ist sehr schwierig, einen passenden Wert für die Anzahl der `whiteLines` zu bestimmen, die die Wortgrenzen markieren. Dies ist z.B. in der letzten Zeile zu erkennen („esponendo la penosissima condizio“). Die Leerstelle zwischen „la“ und „penosissima“ wurde nicht erkannt und falsch segmentiert. Das Problem hier ist, dass bei einer auto-

⁴⁵³ Zur Positionierung der Wörter spielen erst einmal nur die Wortanfänge eine Rolle. Deshalb ist hier jeweils nur der linke Rand der Boundingboxes eingezeichnet, der einen Wortanfang markiert.

matischen Positionierung anhand der so ermittelten Wortgrenzen nicht nur das Wort an dieser Stelle, sondern auch alle nachfolgenden Wörter falsch positioniert würden.⁴⁵⁴

Denkbar wäre, für die Verbesserung solcher Folgefehler wiederum einen manuellen Bearbeitungsmodus zu aktivieren, in dem die Grenzen verschoben und falsche Segmentierungen gelöscht werden könnten. Dies würde allerdings einen erheblichen Arbeitsaufwand für den Editor bedeuten. Elaboriertere Algorithmen sollten hier Abhilfe schaffen.⁴⁵⁵ Ein auf der centerLine eingefügter Text ohne einzelne Wortsegmentierung, aber mit eingerücktem variablem Zeilenbeginn ist in Abbildung 3.29 zu sehen. Wie man an Zeile 5 erkennt, kann es zu größeren Ungenauigkeiten der Zeilenüberblendungen kommen, wenn einzelne Zeichenlängen des verwendeten Schrifttyps nicht denen der Handschrift entsprechen oder die Handschrift in diesem Bezug einen stark variierenden Charakter hat.

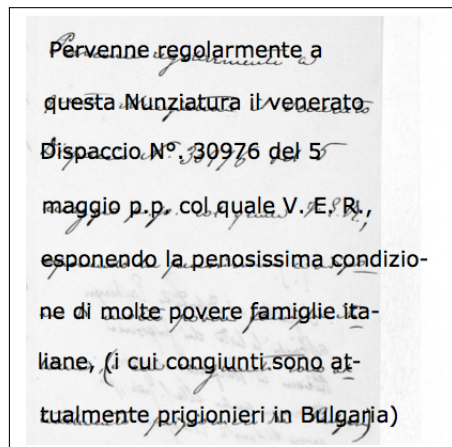


Abbildung 3.29

Um zumindest eine exakte Zeilengenauigkeit mit gleicher Länge zu erreichen, könnten, wie im vorigen Kapitel beschrieben, die Zeichenlängen über Char- und Wordspace konfiguriert werden. Hierzu wäre allerdings noch eine Routine zu implementieren, die einzelne Zeilen formatierbar macht. Dafür müsste die Länge der Zeilen-Boundingbox (dann auch mit exakter Ermittlung des Zeilenendes), die Anzahl der Zeichen der einzufügenden Zeile und die Längen der einzelnen Zeichen des SVG-Schrifttyps ermittelt werden und die Verteilung der Zeichen anhand dieser Parameter neu berechnet werden. Leider existiert keine javascript-Funktion, die die spezifischen Zeichenlängen eines SVG-Schrifttyps ermitteln kann. Am einfachsten wäre es natürlich, das `textLength`-Attribut des SVG-Attributes zu nutzen, um die Berechnung dieser Werte der SVG-Implementation des jeweiligen Browsers zu überlassen. Allerdings wird auch diese nicht browserübergreifend unterstützt.⁴⁵⁶ Die Methoden zum automatischen Einfügen über die Linedetection-Funktionen leisteten bei den Beispielmanuskripten der Pacelli-Edition gute Dienste. Allerdings sind sie nur ein erster Ansatz und sollten durch elaboriertere Algorithmen ergänzt werden. In den letzten Jahren sind einige brauchbare Ansätze in diesem Bereich entwickelt worden. Neben den schon in Kapitel 1.2.4.2 beschriebenen Algorithmen zur Texterkennung scheinen vor allem neuere Ver-

⁴⁵⁴ Vgl. dazu auch: FEINEIS: Annotation (wie Anm. 162).

⁴⁵⁵ Allerdings erscheint es fraglich, ob in Anbetracht der oftmals extrem schwierig zu segmentierenden Handschriften in absehbarer Zeit Algorithmen mit einer so hohen Treffergenauigkeit entwickelt werden, die einen vollautomatischen Segmentierungsprozess zulassen.

⁴⁵⁶ Wie schon beim `Kerning`-Attribut wird auch `textLength` derzeit nur von Safari umgesetzt. Hier bleibt nur zu hoffen, dass die komplette SVG-Spezifikation in Zukunft von allen Browsern implementiert wird.

fahren, die sich speziell mit Textneigungen und der Ermittlung der base line von Zeilen und Wörtern⁴⁵⁷ befassen, interessant zu sein. Es bleibt aber abzuwarten, ob sich diese Algorithmen in einem browserbasierten Umfeld bewähren, da die Berechnung der zu diesen Verfahren benötigten Histogramme sehr aufwendig und rechenintensiv ist.

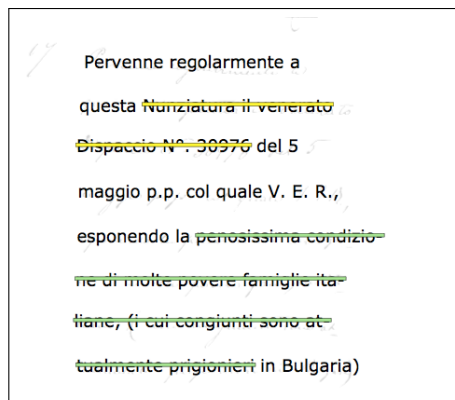


Abbildung 3.30

3.3.4 Streichungen

Wie Text auf dem Digitalisat positioniert werden kann, ist in vorigem Abschnitt ausführlich beschrieben worden. Die zweite zentrale Komponente textgenetischer Operationen ist das Entfernen von Text (Streichungen, gänzliche Tilgungen, etc.). Für das Streichen werden wieder die im Editor ausgezeichneten Koordinaten der Linewidgets übernommen (lineStart, lineEnd, charStart, charEnd) und damit die entsprechenden Textelemente der SVG-View mit den gewünschten Formatierungen versehen (fiktives Streichungsbeispiel in Abbildung 3.30). Für die Streichung

wurde ein einfaches SVG-rect-Element implementiert, dass zusätzlich die über die Konfigurationsdatei erhältlichen Parameter der Schreiber-Level (hier unterschiedliche Farben der Streichungen für unterschiedliche Bearbeiter) verwendet. Die Streichungen wurden auf der Mittellinie der Boundingboxen der jeweiligen Zeilen positioniert. Hier macht sich wieder einmal der Vorteil der stand-off-basierten Eingabe gegenüber einer Kodierung direkt in XML bemerkbar. Trotz überlappender Textsegmente (Zeilen- und Spaltenumbrüche), muss die Kodierung der Streichung nicht extra fragmentiert oder durch Milestonelösungen simuliert werden.

Es spielt in der SVG-View keine Rolle, ob der Zeilenabstand der Spalten unterschiedlich groß ist oder ein Textteil rotiert und der andere nicht, die Formatierungselemente richten sich immer an den Koordinaten der jeweiligen Zeile aus (Abbildung 3.31). Dies geschieht, indem sie z.B. bei der Zeilenneigung die Rotationseigenschaft von ihrem Parentelement erben. Die Struktur eines solchen Textelements kombiniert mit einer Streichung ist in Listing 3.14 zu sehen. Im Rotelement ist im transform-Attribut der Rotationswinkel definiert, der automatisch an alle Kindelemente vererbt wird. Auf diese Art und Weise lassen sich im Prinzip beliebige SVG-Elemente definieren und in die SVG-View integrieren. Dies könnten z.B. Unterstreichungen in unterschiedlichen Formen (gestrichelt, punktiert) sein, verschiedene Metamarkie-

⁴⁵⁷ Vgl. hierzu auch: Kamil R. AIDA-ZADE/Jamaladdin Z. HASANOV: Word base line detection in handwritten text recognition systems, in: International Journal of Electrical and Computer Engineering 4.5 2009, S. 310–314 und Houcine BOUBAKER/Monji KHERALLHA/Adel M. ALIMI: New Algorithm of Straight or Curved Baseline Detection for Short Arabic Handwritten writing, in: 10th International Conference on Document Analysis and Recognition 2009, S. 778–781.

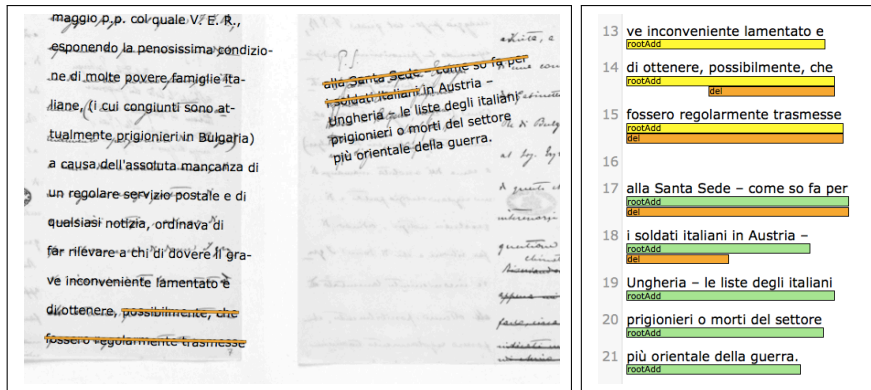


Abbildung 3.31 Streichung über verschiedene Zeilen- und Spalten mit unterschiedlichem Zeilenabstand und Rotationswinkel. links: SVG-View, rechts: Editor

rungssymbole oder auch Tooltips, die an der jeweiligen Zeichenposition textkritische Bemerkungen einblenden. Dafür wären lediglich weitere Elemente in der Elementslist (wie in Abschnitt 3.2.8 beschrieben) anzulegen, ein beliebig gestaltbares SVG-Element zu definieren und anzugeben, wo dieses positioniert werden soll, z.B. relativ zu den x und y-Koordinaten von einem der 4 Eckpunkte der Boundingbox einer ausgezeichneten Textpassage. So definierte und in einer Online-Bibliothek zur Verfügung gestellte SVG-Grafikelemente könnten dann auch dynamisch geladen und mit einem entsprechenden Skalierungsfaktor in die jeweilige diplomatische Ansicht integriert werden.

```

1 <g id="rootA_PolyText_3" inclination="rotate" x="2171" y="326" transform="
  rotate(28.422185568666787, 2141, 316)" visibility="visible">
2   <text x="2171" y="326" dxVal="0" wSpace="0" font-size="38" font-family="
    Verdana" ln="root_ln_16">alla Santa Sede -come so fa per </text>
3   <text x="2171" y="477.7969449626704" dxVal="0" wSpace="0" font-size="38"
    font-family="Verdana" ln="root_ln_17">i soldati italiani in Austria -<
    /text>
4   ...
5   <g delID="del_line_11_2">
6     <rect id="rectClone" x="2168.46484375" y="306.5764973958333"
        stroke-width="1" fill="rgb(144, 238, 144)" width="639.23046875"
        height="8.444661458333334">
7     <rect id="rectClone" x="2168.46484375" y="458.3733723958333"
        stroke-width="1" fill="rgb(144, 238, 144)" width="526.7734375"
        height="8.444661458333334">
8     ...
9   </g>
10 </g>

```

Listing 3.14

3.4 Textgenese dynamisch visualisieren

3.4.1 Navigationsbaum

Was nun noch fehlt, ist ein Instrument, mit dem sich die einzelnen Schritte der Textgenese in ihrer jeweiligen Abfolge visualisieren lassen. Dazu dient uns das in Kapitel 2.5.2 beschriebene kaskadierende Zählersystem der Level. Die Untergliederung der textgenetischen Schichten und Einzelprozesse in beliebig tief verschachtelbare Einheiten ist ideal geeignet, um ein baumartiges Navigationsinstrument zu implementieren. Über das Menü Tree-View gelangt man zu einer solchen Navigation (Abbildung 3.32). In der Baumstruktur werden die einzelnen Bearbeitungsschritte sortiert nach den Level, die während der Bearbeitung des Textzeugen vergeben wurden, angezeigt. Jede Untergliederung eines Level in einen Sublevel wird durch einen expandierbaren Knotenpunkt im Baum symbolisiert. In der Grundeinstellung ist in der SVG-View nur das Digitalisat des Textzeugen eingeblendet. In den Editorinstanzen sind alle Linewidgetmarkierungen inaktiv, werden also in ihrer jeweiligen Farbe (wenn eine vergeben wurde) angezeigt. Beim Anklicken eines der Blätter oder Äste werden die entsprechenden Textelemente im Editor aktiviert (rotes Markup-Widget) und die jeweiligen Aktionen in der SVG-View ausgeführt (Einfügen oder Streichung eines Textsegmentes etc., siehe Abbildung 3.33). Durch einen Mausklick auf ein Baumblatt wird eine einzelne Aktion ausgeführt, bei einem Klick auf einen Astknoten, alle dort kodierten Aktionen der Sublevels. (Es ist nicht möglich eine del-Funktion auszuführen, bevor nicht die vorherige add-Funktion ausgeführt wurde) Die Untergliederung der Level hat den Vorteil, dass man sich sowohl Geneseschichten, die aus komplexeren Modifikationen bestehen (z.B. gebündelte Bearbeitungsschritte eines Schreibers), auf einmal als auch die Modifikationen einzeln anzeigen lassen kann.

Somit können nun auch Schreibprozesse auf einer granular sehr feinen Ebene betrachtet werden, wie an folgendem Beispiel aus einem Entwurf zu einem Nuntiatursbericht Eugenio Pacellis zu sehen ist. Verfasst wurde dieser von dem Sekretär Lorenzo Schioppa und anschließend von Pacelli selbst überarbeitet. Die Ausgangssituation zu Beginn der Visualisierung ohne selektiertem Knoten in der Treeview stellt sich wie folgt dar (Abbildung 3.34): Das Digitalisat ohne Texteinblendung ist zu sehen. Im Editorfenster sind die Widgets nicht aktiviert (gelbe Widgets kennzeichnen die Geneseschicht von Schioppa, grüne die Änderungen von Pacelli). Nach dem Auswählen der kompletten ersten Geneseschicht (Grundtext von Schioppa), wird der Text topographisch genau auf dem Digitalisat positioniert und die Linewidgets der Schioppa-Schicht im Editor rot markiert.

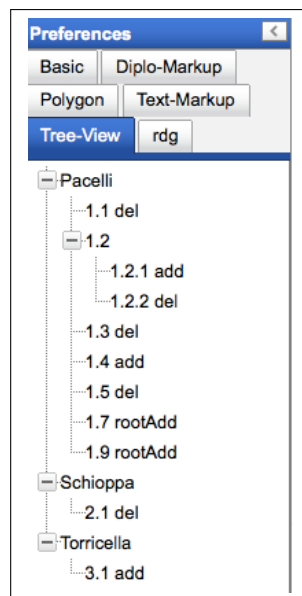


Abbildung 3.32

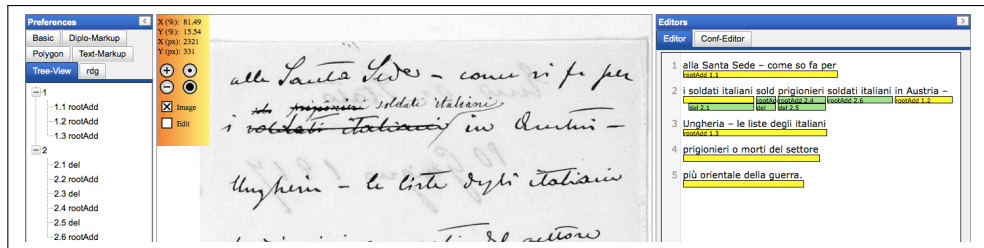


Abbildung 3.34 Prigionieri 1

Somit lässt sich erkennen, welche Schreibprozesse in der diplomatischen Ansicht schon visualisiert worden sind (Abbildung 3.35). Die erste Modifikation von Pacelli ist das Streichen der Textstelle 'soldati italiani' (Abbildung 3.36). Anschließend beginnt er, den gestrichenen Text über der Zeile erneut mit 'soldati' zu ersetzen, bricht aber nach 'sold' ab (Abbildung 3.37), streicht es (Abbildung 3.38) und setzt als Sofortkorrektur statt dessen 'prigionieri' (Häftlinge) ein (Abbildung 3.39). Auch dieses streicht er (Abbildung 3.40) und fügt nun wieder den ursprünglichen Text 'soldati italiani' ein (Abbildung 3.41).

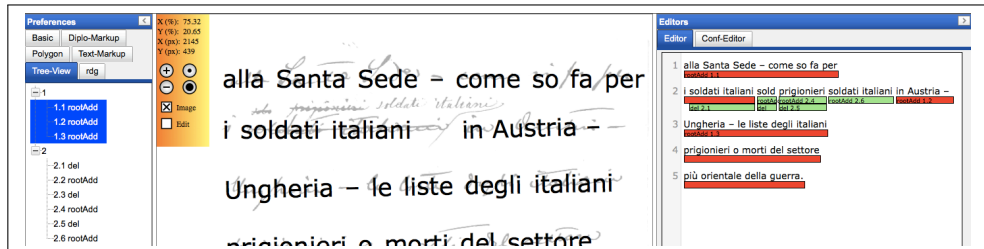


Abbildung 3.35 Prigionieri 2

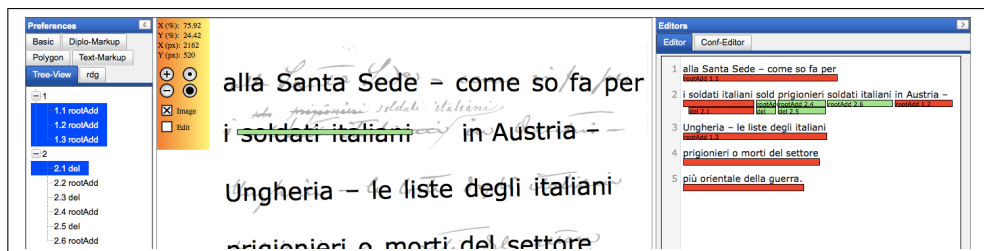


Abbildung 3.36 Prigionieri 3

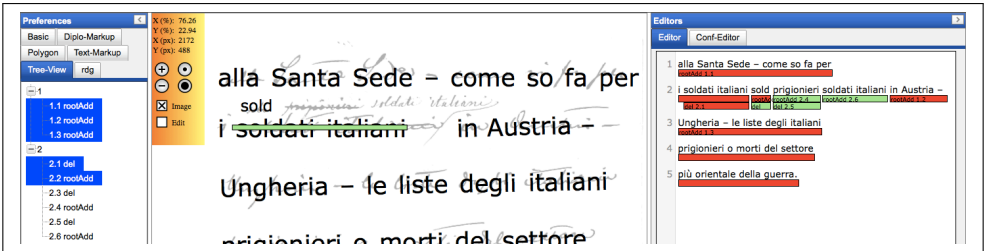


Abbildung 3.37 Prigionieri 4

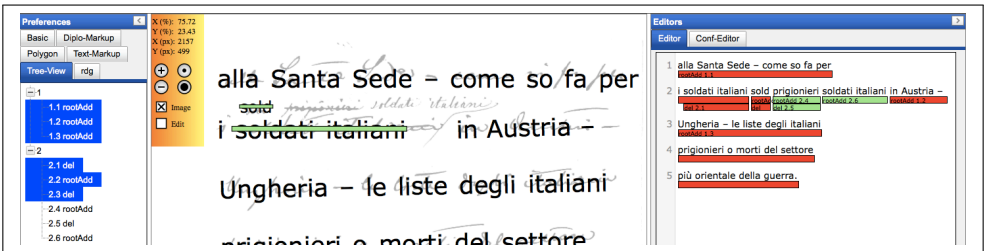


Abbildung 3.38 Prigionieri 5

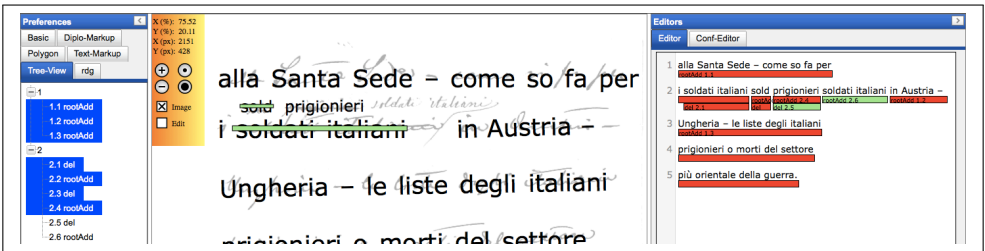


Abbildung 3.39 Prigionieri 6

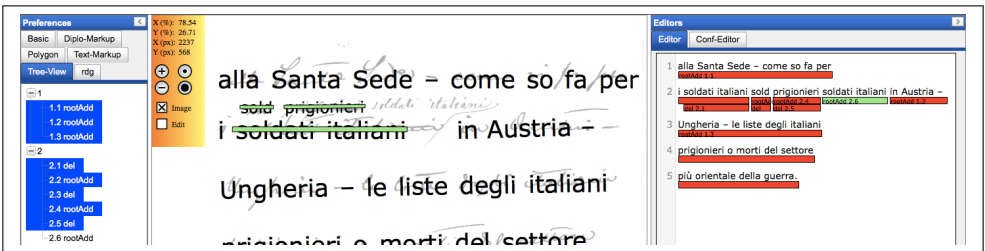


Abbildung 3.40 Prigionieri 7

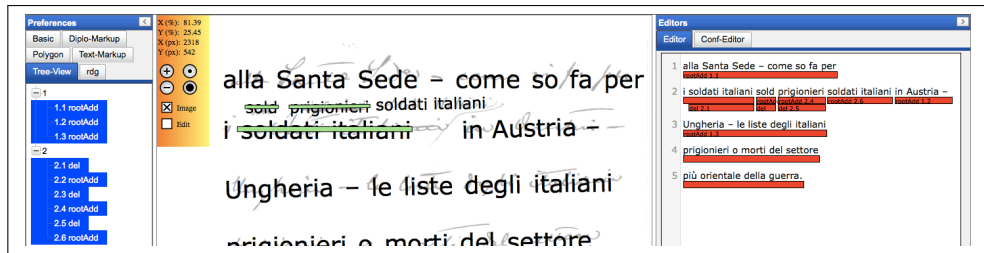


Abbildung 3.41 Prigionieri 8

Die vorgestellte Transkription basiert auf dem inline-Verfahren der rootAdd-Methode, nutzt also zur Kodierung allein die Haupteditorinstanz. Die von Pacelli eingefügte Textzeile ist dort genau wie bei den gängigen Lösungsansätzen des Musil- und Wittgenstein-Beispiels direkt im linearen Textfluss integriert. Zu dem gleichen Er-

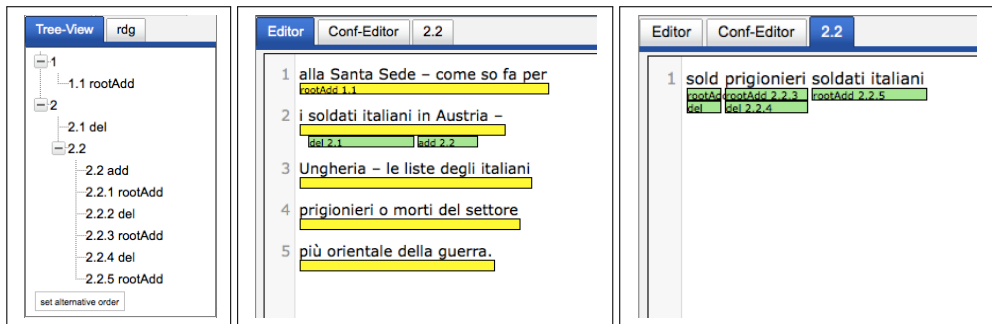


Abbildung 3.42

gebnis (was die diplomatische Darstellung der SVG-View betrifft) hätte man auch gelangen können, wenn man die add-Methode und somit die Darstellung in multiplen Editorfenstern bevorzugt hätte. Hier wäre an der Stelle der Einfügung ein add notiert worden und die einzelnen Prozesse hätten dann separat im Editorfenster 2.2 kodiert werden können (Abbildung 3.42). Mit dieser Variante erreicht man eine saubere Trennung der verschiedenen Geneseschichten in Bezug auf die Texthinzufügungen. Die Streichungen lassen sich so natürlich nicht separieren, da sie ja auch unmittelbar zwei Schichten betreffen können, wie in diesem Beispiel die erste Streichung Pacellis einer Textpassage von Schioppa 'soldati italiani' verdeutlicht. Zusätzlich hätte der Prozess im Genesebaum auch als eigener Knoten kodiert werden können (2.2.2). Auf diese Art und Weise lassen sich komplexere Modifikationsprozesse bündeln und im Navigationsbaum übersichtlicher anzeigen, indem die Knoten mit den komplexeren Prozessen einfach eingeklappt werden. Man erhält dadurch eine überblicksartige Darstellung der Veränderungen am kompletten Dokument auf einer höheren Hierarchieebene. Bei Bedarf könnten die einzelnen Prozessknoten dann aufgeklappt und genauer analysiert werden.

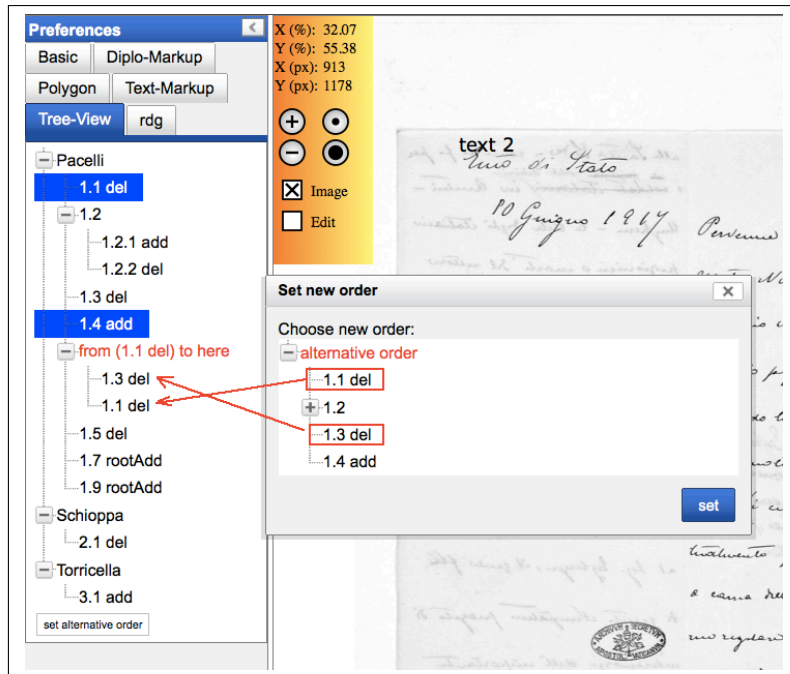


Abbildung 3.43

3.4.2 Chronologische Lesarten hinzufügen

Da die Rekonstruktion der Textgenese zu unterschiedlichen chronologischen Abfolgen führen kann (z.B. bei der Produktions- und Rezeptionschronologie, siehe Abschnitt 2.5), und da diese im Prinzip offen für weitere Interpretationen von Abfolgen sein müssen, besteht die Möglichkeit, neue alternative Reihenfolgen in diesem Baum anzulegen. Dazu werden innerhalb des Navigationsbaumes die Knotenpunkte markiert, die eine alternative chronologische Abfolge erhalten sollen, und der Button 'set alternative order' betätigt. Daraufhin öffnet sich ein Fenster mit genau diesem neu zu sortierenden Baumausschnitt (Abbildung 3.43). Gleichzeitig wird an der Stelle hinter dem markierten Baumausschnitt ein neuer leerer Knoten angelegt mit der Angabe, für welchen Baumausschnitt diese Alternative Gültigkeit besitzt. In dem geöffneten Fenster kann nun durch einfaches Anklicken eines Knotens oder Blatts dieser/dieses in den Alternativknoten im Genesebaum eingefügt werden. Die Knoten und Blätter werden dabei einfach in der Reihenfolge des Anklickens eingegliedert. Der Vorteil bei diesem Verfahren liegt darin, dass ganze Sublevelknoten neu einsortiert werden können ohne die Reihenfolge ihrer Kindelemente zu verändern. Das kann sehr sinnvoll sein, wenn die Knoten auf einer höheren Levelhierarchie zwar umsortiert werden sollen, z.B. ganze Textblöcke, die aus zahlreichen einzelnen Modifikationsprozessen von (intralinen) Einfügungen und Streichungen bestehen, die Abfolge dieser Modifikationen selbst aber erhalten bleiben soll. Ein Beispiel dafür könnte eine Neusortierung

der Textblöcke des Musil-Manuskriptes in Abbildung 2.16 sein. Die alternativen Gruppierungen können im vorliegenden System separat verwaltet werden, da diese nicht direkt mit der Grundtranskription in Verbindung stehen müssen. Im Allgemeinen sollte angestrebt werden, die Basistranskription von möglichen Alternativtranskriptionen externer Nutzer zu trennen,⁴⁵⁸ weshalb die Alternativsortierungen in dieser Anwendung auch separat in der Konfigurationsdatei verwaltet werden:

```

1 <sortorder>
2   <alt from="1.2" to="1.7">
3     <item clone="1.2"/>
4     <item clone="1.5"/>
5     <item clone="1.4"/>
6     <item clone="1.3"/>
7     <item clone="1.7"/>
8   </alt>
9   <alt>
10    ...
11  </alt>
12 </sortorder>

```

Listing 3.15

In dieser Struktur geben *from* und *to* an, zu welchem Level-Range eine Alternative vorliegt. Die Reihenfolge der *item*-Elemente bestimmt die Neusortierung der im *clone*-Attribut verzeichneten Knoten. Beim Laden der Tree-View werden diese dann aus der Grundkodierung geclost und als alternative Abfolge an der jeweiligen Stelle in den Baum der Grundtranskription eingefügt. Auf diese Art und Weise lassen sich die in Kapitel 2.5.2 beschriebenen unterschiedlichen Chronologien der Textproduktion und -rezeption und andere interpretative chronologische Varianten visualisieren. Das Digitalisat kann über das Auswahlfeld *Image* im Navigationspanel der SVG-View ein- und ausgeblendet werden. Soll das Digitalisat nur aufgehellt werden, um die Lesbarkeit des überblendeten Textes zu verbessern, sollen aber gleichzeitig die Konturen anderer Bildsegmente weiterhin erkennbar bleiben, kann im Basic-Menü die Helligkeit angepasst werden.

Auch die in Abschnitt 2.5.2 beschriebenen chronologischen Lesarten lassen sich so im Sinne Lebraves visualisieren. Die Sequenzierung und das anschließende Anlegen von alternativen Lesarten bietet dem Nutzer dann die Möglichkeit, verschiedene Wege durch die Textgenese zu beschreiten. In Abbildung 3.44 sind alle identifizierten Lesarten des 'weiße Schloß'-Beispiels (Abbildung 2.22) in ihrer alternativen chronologischen Abfolge zu sehen (in der Mitte die diplomatische Ansicht, rechts die jeweils dazugehörigen aktivierten Widgets).

Alle Komponenten dieser Applikation können auch ohne jegliche Editionsfunktion

⁴⁵⁸ Vgl. SAHLE: Digitale Editionsformen 3 (wie Anm. 54), S. 250.

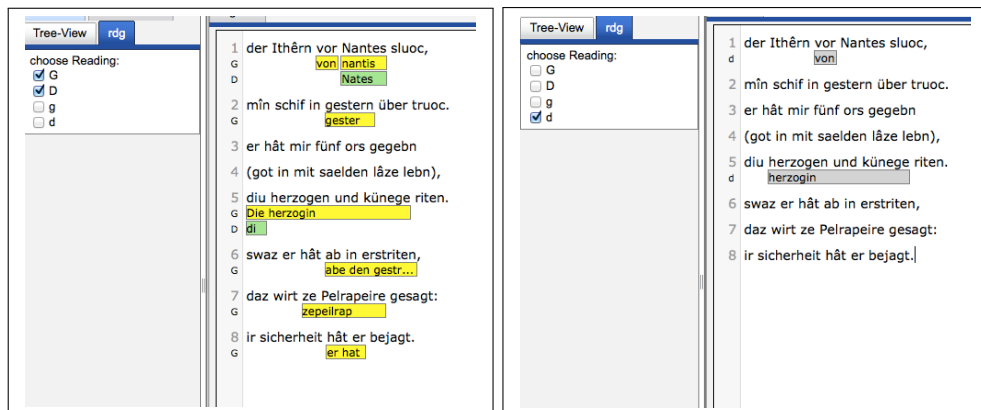


Abbildung 3.45

cher Handschriften, ohne alle Textzeugen nebeneinander öffnen zu müssen, wie dies z.B. bei der in Kapitel 2.6 beschriebenen Versioning-Machine der Fall ist. Beliebige Handschriften können zeilenweise untereinander eingeblendet werden und somit lassen sich Übereinstimmungen und Abweichungen rasch auffinden. Für die Parzival-Strophe aus Abschnitt 3.2.10 z.B. ist das enge Verwandtschaftsverhältnis der beiden Handschriften G und g bei direkter Gegenüberstellung gut zu erkennen (Abbildung 3.46).

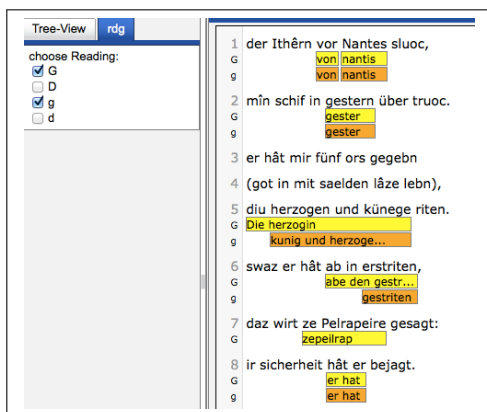


Abbildung 3.46

3.4.4 Visualisierung über HTML

Aus dieser Anwendung heraus können auch dynamische Prozessvisualisierungen implementiert werden, die sich weder einer diplomatischen Ansicht noch der Widget-Objekte bedienen, sondern auf klassische Weise formatierten HTML-Output produzieren. Die Einzelkomponenten der Applikation sind hier in einer etwas anderen Art und Weise kombiniert worden. Um das Digitalisat mit dem Text zu verknüpfen, wurde die einfachere Komponente des Polygonlayers verwendet. Die

Tree-View (hier Gen-Tree genannt) erzeugt in dem mittleren HTML-Fenster eine Ausgabe, die auf eine prozedurale Art und Weise die Textentstehung nachvollziehbar macht. Abbildung 3.47 zeigt die unterschiedlichen Geneseschichten, die durch den mouseover-Effekt des Gen-Tree dynamisch eingeblendet werden. Um die einzelnen Bearbeitungsschritte innerhalb einer Schicht anzuzeigen, kann man den Ast der Ge-

neseschicht aufklappen und sich die gewünschten Bearbeitungsschritte einzeln anzeigen lassen. Dabei wird die betreffende Textstelle hervorgehoben, der textkritische Kommentar eingeblendet und sofern über polyID verlinkt, der betreffende Bildausschnitt des Digitalisats angezeigt.

In diesem HTML-Modus lassen sich nun auch die in Abschnitt 3.2.5 beschriebenen Transpositionen über physische Dokumentgrenzen hinweg prozedural einblenden. In Abbildung 3.48 ist wieder der Genesebaum zu sehen, der unter dem Level 3.1 eine Abfolge von Umstellungen verzeichnet (hier eine komplexe Neusortierung von Listeneinträgen). Die Transpositionen lassen sich einzeln über die row-Knoten (Abbildung 3.48 Mitte) oder komplett über den all-Knoten ausführen (Abbildung 3.48 unten). Eine gleichzeitige Darstellung beider Schriftstücke (Entwurf und Ausfertigung) in diplomatischer Form oder mit verlinkten Bildausschnitten würde den Transformationsprozess auch hier noch deutlicher zutage treten lassen. Da zu diesem Entwurfsbeispiel aber keine Ausfertigung mehr existiert, erscheint es hier umso sinnvoller, den Transformationsprozess dann wenigstens auf diese Art und Weise zu visualisieren.

4 Fazit/Ausblick

Im Jahr 2001 resümierte Sören Steding die Ergebnisse einer Studie über die Benutzererwartungen an Digitale Editionen wie folgt: „Der Wunsch nach textgenetischen Informationen und Faksimiles ist insgesamt gering. Dies lässt schließen, dass die Mehrheit der Teilnehmer weder Textkonstitution betreiben will, noch daran interessiert ist, den edierten Text und die Entscheidungen der Editoren anhand von Abbildungen und Faksimiles zu verifizieren.“⁴⁶⁰ Offen bleibt in dieser Studie, ob dieses Desinteresse auf allgemeine Vorlieben und Abneigungen der Probanden zurückzuführen war oder ob die technologischen Möglichkeiten damaliger Digitaler Editionen nicht den Ansprüchen der Probanden entsprachen. Einen Anhaltspunkt für letzteres könnte die Studie 'Revolution or Remediation'⁴⁶¹ von Karlsson und Malm geben, die als Ergebnis zu dem Schluß gelangt, dass eine wirkliche Revolution der Editionswissenschaft innerhalb des digitalen Mediums eben nicht stattgefunden habe und dass sich die traditionellen Methoden und Theorien nur in das neue Medium verlagert hätten, ohne wirklich zu neuen Innovationen geführt zu haben.

Nach Auffassung des Autors hat dieses mangelnde Interesse insbesondere an textgenetischen Fragestellungen zu großen Teilen damit zu tun, dass die Möglichkeiten Digitaler Editionen in browserbasierten Umgebungen noch nicht konsequent genug ausgeschöpft wurden. Auch wenn seit der letztgenannten Studie fast 10 Jahre vergangen sind, hat sich an der Grundsituation nicht viel geändert.⁴⁶² Wie die Beispiele in Abschnitt 2.4 zeigen, sind diplomatische Transkriptionen nach wie vor bemüht, genetische Aspekte in zweidimensionalen linearen Texten abgetrennt vom eigentlichen Original (Faksimile) darzustellen und ahmen im Prinzip nur die schon zu Zeiten der Druckkultur gängigen Editionsmethoden nach.

In der vorliegenden Arbeit wurde gezeigt, dass andere Konzepte der Kodierung und Visualisierung notwendig sind, um diese Probleme zu lösen. In dem hier vertretenen Textverständnis dient der eigentliche Auszeichnungstext in den Editorbereichen nur

⁴⁶⁰ Sören A. STEDING: Benutzerorientierte Digitale Editionen: Eine empirische Annäherung, in: www.germanistik2001.de. Vorträge des Erlanger Germanistentags 2002, hrsg. v. Hartmut KUGLER, S. 729–742, hier S. 733. Durchaus kritisch zu betrachten ist hier allerdings die Auswahl der Probanden, die vornehmlich aus Studenten der Germanistik und Anglistik bestanden. Die Schlussfolgerung kann also nicht als repräsentative Stellungnahme für die Gesamtheit der Geisteswissenschaften gelten. Die Fragebögen scheinen auch eher auf die Vorteile Digitaler Editionen in Bezug auf Themenbereiche und Fragestellungen einzugehen, die noch stark von der Sichtweise und den Möglichkeiten klassischer Printeditionen bestimmt waren.

⁴⁶¹ Lina KARLSSON/Linda MALM: Revolution or Remediation? A Study of Electronic Scholarly Editions on the Web, in: *Human IT* (7.1) 2004, S. 1–46.

⁴⁶² Aktuellere Studien die sich speziell mit den Benutzungsgewohnheiten textgenetischer Digitaler Editionen befassen, sind dem Autor dieser Arbeit leider nicht bekannt und bilden hier noch ein Forschungsdesiderat.

noch der temporären Verwaltung zum Zweck der Informationskodierung und des Informationretrievals. Die eigentliche Darstellung der verschiedenen chronologischen und räumlichen Ebenen und Prozesse lässt sich aber viel anschaulicher und überprüfbarer mittels Konzepten wie der hier vorgestellten Möglichkeit, den Text dort einzubinden, wo er den größten Bezug zum Faksimile hat, realisieren, da die Schreibräume genau dort angeordnet werden können, wo sie auch im Original zu finden sind. Somit besteht gar nicht mehr die Notwendigkeit, symbolhafte HTML-Formatierungen und diakritische Zeichen für eine reichlich kryptische und ungenaue Informationssimulation zu verwenden. Die Transferleistung, die der Benutzer der Edition bei der Überführung von Textsequenzen von der Transkription auf das Digitalisat bisher zu erbringen hatte, entfallen somit. Die eigentlichen topographischen Informationen ergeben sich aus den tatsächlichen Positionierungen der Textsegmente und der Möglichkeit, unterschiedliche Formen der Streichungen, Unterstreichungen, Metamarkierungen etc. mit graphisch eindeutigen SVG-Elementen zu symbolisieren. Somit fallen Transkription und Faksimile aus Benutzerperspektive letztendlich zusammen und bilden wieder eine schon seit langem in der Editionswissenschaft geforderte Einheit. Daneben sollte die Fokussierung auf einen dynamischen Textbegriff, der variabel definierbare chronologische Abläufe, die daraus resultierenden Kombinationsmöglichkeiten von Textfragmenten und die prinzipielle Offenheit für neue Interpretationen unterstützt, sollte der Vision der frühen Hypertextgeneration von einem offenen, kollaborativen, nicht-linearen Textverständnis Rechnung tragen.

Diese Sicht auf den Text wirft Fragen bezüglich der Zitierbarkeit des Textes auf (oder besser der Texte im Kontext von Raum und Zeit). Eine instabile Zitierweise ist dabei die eines wie immer konstituierten (durch Deutung 'kontaminierten') linearen Textes. Dieser kann im Laufe der Zeit immer wieder Veränderungen unterliegen. Eine weitaus stabilere Form der Zitierung ist die, die sich auf das Original (oder besser die Koordinaten eines digitalen Faksimile) bezieht. Der transkribierte Text ist dann auch hier nur noch temporäre Ausdrucksform und eine mögliche Repräsentation von vielen. Das Original war und wird immer im Mittelpunkt des Forschungsinteresses stehen. Deshalb ist es auch von zentraler Bedeutung, dass Wissenschaftlern Plattformen und Systeme zur Verfügung gestellt werden, die den Austausch von Hypothesen unterstützen, die den aktuellen Stand der Forschung widerspiegeln und die Argumentationen innerhalb der dort geführten Diskussion nachvollziehbar machen. Hier fehlt es derzeit noch an geeigneten Plattformen, um diesen lebendigen Dialog zu fördern. In solchen Foren wird es immer wichtiger werden, über welches Textphänomen gerade diskutiert wird, ein einfaches Abbild des Originals wird in vielen Fällen nicht ausreichen. Es muss granular feiner auf Textphänomene innerhalb des Originals verwiesen werden können. Solch ein Diskussionsprozess muss immer ein offener sein, der wohl nie den Status der Abgeschlossenheit erlangen dürfte. Deshalb müssen Diskurse zu jeder beliebigen Zeit zitierbar sein und auch bleiben. Versionierungssysteme, die frühere Diskussionsstände speichern, werden hier in Zukunft größere Bedeutung erlangen.

Auch die Konfigurierbarkeit und Erweiterbarkeit von Systemen und die Modularität (im Sinne von Microservices⁴⁶³ z.B.) sind hier nicht zu vernachlässigende Fak-

⁴⁶³ Joris van ZUNDERT: If you Build it, Will We come? Large Scale Digital Infrastructures as a Dead

toren. Gleiches gilt für die Offenlegung und Dokumentation von Algorithmen und Programmarchitekturen. Dieses erscheint in Anbetracht der oft heterogenen Voraussetzungen und Anforderungen von Editionsprojekten einen wesentlichen Mehrwert zu versprechen. Neue Projekte können in der Regel bis zu einem bestimmten Grad von zuvor geleisteter Entwicklungsarbeit profitieren, aber nur in den seltensten Fällen die Methoden, Algorithmen und Frameworkarchitekturen eins zu eins übernehmen. Jede Edition hat ihre technischen Besonderheiten und spezifischen Fragestellungen. Das größtmögliche Potential lässt sich aber nur dann ausschöpfen, wenn keine vorgefertigten Modelle und Standards übernommen werden müssen, die nicht für diesen Zwecke geeignet sind. Standardisierte Schnittstellen, die einen Austausch von Funktionalität gewährleisten, sind mindestens ebenso wichtig wie die Auszeichnungsstandards selbst.⁴⁶⁴ In der Zukunft sollte es möglich sein, sich in einer Art Baukastenprinzip seine eigene, auf die spezifischen Projektbedürfnisse angepasste Entwicklungsumgebung zusammenzustellen. Dies könnte z.B. in Form von Web-Applikationen geschehen, die in Reihe geschaltet beliebige Analysevorgänge verbinden und nach Bedarf standardisierte Verarbeitungsalgorithmen nachladen können.

Systeme, wie das hier vorgestellte, könnten auch vermehrt in der Lehre eingesetzt werden. Man stelle sich z.B. vor, dass die obligatorischen Quellenstudien in den klassischen Einführungsseminaren der Geschichtswissenschaft direkt am digitalen Objekt stattfänden und Bestimmungen von beispielsweise Urkunden-Komponenten (Intitulatio, Narratio, Promulgatio, Arrenga...) somit auch über das Ende dieser Seminare hinaus erhalten blieben. Dies hätte einen doppelten Effekt. Zum einen würden Studenten so praktische Erfahrungen im Umgang mit digitalen Werkzeugen sammeln und dort ihre Kernkompetenzen erweitern, zum anderen wären die Ergebnisse in Form von Grundtranskriptionen für die Wissenschaft weiter nutzbar.

In Zukunft wird auch auf die Herausforderungen eines veränderten Kommunikationsverhaltens reagiert werden müssen. Derzeit verlagert sich der größte Teil des gesamten digitalen Datentransfers in rasantem Tempo auf die Ebene mobiler Endgeräte (Laptops, Smartphones, Tablets). Dazu bemerkt der Historiker William Thomas: „This shift to mobile devices deserves our close attention. They are closed systems, and the experience of a self-contained 'app' fundamentally more restricted than the hypertextual World Wide Web. This trend marks a fundamental change that challenges Digital Humanities scholars to consider new modes of production at the same time as it provides enormous opportunities to reach wider audiences and alter the form of our scholarly communication.“⁴⁶⁵ Bisher sind diese Möglichkeiten aber noch so gut wie gar nicht innerhalb der Digital Humanities diskutiert, geschweige denn Konzepte

End for Digital Humanities, in: *Historical Social Research / Historische Sozialforschung* (37.3) 2012, S. 106–124.

⁴⁶⁴ Vgl. hierzu z.B. die About-Seite von Interedition: „Typically, these editions have been made by single projects and are accessible only through dedicated and specialized interfaces. This leads to profound problems of sustainability, as the interfaces grow old, and of interoperability, as the data cannot be addressed except through the project interface.“ http://www.interedition.eu/?page_id=84

⁴⁶⁵ William G. THOMAS: Trends in Digital Humanities: remarks at the CIC Digital Humanities Summit ('the keynote in the dark'), 2012, URL: <http://railroads.unl.edu/blog/?p=794>.

dafür entwickelt worden.⁴⁶⁶ Für Deutschland wird bis 2015 eine jährliche Wachstumsrate des mobilen Datenverkehrs von 97% prognostiziert.⁴⁶⁷ Hierfür verantwortlich ist vor allem das ständig steigende Angebot an Apps und anderen Anwendungen. Diese ermöglichen einen Zugang zu Informationen zu jeder Zeit und vor allem an jedem Ort. Die Herausforderungen bestehen hier unter anderem in immer kleiner werdenden Displays und neuen Eingabemethoden wie z.B. den Touchscreen-Technologien. Auf die Darstellung von Informationen bezogen bedeutet dies, dass die Anordnung von Informationen auf diesen Bildausschnitten und somit auch ihre Reduzierung auf bestimmte Sichten immer wichtiger werden, also auch hier das Modularisieren von Informationsebenen im Mittelpunkt steht. Hier ist eine Tendenz in Richtung modularen light-weight-Komponenten zu beobachten. Mobile Apps bieten zumeist Lösungen für ganz bestimmte Zwecke mit stark applikationsspezifischem Funktionsumfang. Allumfassende Anwendungen in Form von Komplettsystemen bilden eher die Ausnahme. So ist zu erwarten, dass sich in diesem Umfeld auch geisteswissenschaftliche Informationsapps entwickeln werden, die sich auf bestimmte Einzelaspekte konzentrieren. Die Entwicklungen in diesen Bereichen werden mit großer Wahrscheinlichkeit auch zu veränderten Lese- und Benutzungsgewohnheiten führen.⁴⁶⁸

Die Frage stellt sich, wie diesen Herausforderungen in Zukunft insgesamt am besten zu begegnen ist und wie solche Systeme am effizientesten entwickelt werden können, damit das sprichwörtliche Rad nicht immer wieder neu erfunden wird und die notwendige Stabilität, Nachnutzbarkeit und Langzeitverfügbarkeit gewährleistet werden kann.⁴⁶⁹ Es wird hier weniger darauf ankommen, ein generalisiertes, schwerfälliges Standardisierungsgebäude aufzubauen. Die Heterogenität geisteswissenschaftlicher Informationen und Fragestellungen wird wahrscheinlich nie zu einem alles umfassenden Datenstandard führen, der allen Ansprüchen genügt und gleichzeitig so streng normativ ausgerichtet ist, dass Softwaresysteme in der Lage sind, alle möglichen Methoden und Funktionen auf Grundlage dieser Standards zu implementieren.

Große Initiativen zum Aufbau von Forschungsinfrastrukturen führen hier nicht selten in eine Sackgasse, wie schon Joris van Zundert zutreffend analysiert hat.⁴⁷⁰ Dies ist auch nicht verwunderlich, wenn große Teile der Fördermittel in Organisationsstrukturen fließen und 'Überstandardisierungen und -generalisierungen' sich innovationshemmend auswirken. So wie bei einer 'Revolution von unten' entstehen innovative Produkte meist in kleineren Entwicklerteams direkt an der Forschungsbasis. Neuere Initiativen, wie sie z.B. Interedition⁴⁷¹ darstellt, scheinen hier vielversprechende Ansätze zu bieten. Hier können engagierte Softwareentwickler aus dem Bereich der

⁴⁶⁶ Vgl. Manfred THALLER: Controversies around the Digital Humanities: An Agenda, in: *Historical Social Research / Historische Sozialforschung* (37.3) 2012, S. 7–24.

⁴⁶⁷ Cisco Systems Inc. (O.V.): Global Mobile Data Traffic Forecast Update, 2010-2015, in: Cisco Visual Networking Index 2011, URL: newsroom.cisco.com/dlls/ekits/Cisco_VNI_Global_Mobile_Data_Traffic_Forecast_2010_2015.pdf.

⁴⁶⁸ Vgl. Amelia SANZ/Maria GOICOECHEA: Literary reading rituals and practices on new interfaces, in: *Journal of the Alliance of Digital Humanities Organizations*. (27.3) 2012.

⁴⁶⁹ Wobei es hier nicht nur um die Artefakte des Kulturerbes an sich, sondern ebenso um die Werkzeuge und ihre Funktionalität gehen muss.

⁴⁷⁰ ZUNDELT: Infrastructures (wie Anm. 463).

⁴⁷¹ <http://www.interedition.eu>

Digital Humanities in eigens organisierten Bootcamps, die sich mit Teilbereichen geisteswissenschaftlicher Informationsverarbeitung befassen, ohne den Standardisierungsdruck großer Forschungsinitiativen Systeme und Methoden entwickeln und in Form von modularen Systemen und Microservices zur Verfügung stellen.

Literaturverzeichnis

- AIDA-ZADE, Kamil R. und Jamaladdin Z. HASANOV: Word base line detection in handwritten text recognition systems, in: International Journal of Electrical and Computer Engineering 4.5 2009, S. 310–314.
- ASCHENBRENNER, Andreas und Katja MEFFERT: Wissenschaftliche Infrastruktur in den Geisteswissenschaften? – Eine Wegbeschreibung, in: Jahrbuch für Computerphilologie 9 2009, URL: <http://computerphilologie.digital-humanities.de/jg07/aschmeff.html>.
- ASSMANN, Bernhard: Sind die kanonischen Zitierweisen der Geisteswissenschaften als nachhaltige Komponenten digitaler Repositorien geeignet? (Magisterarbeit), Universität zu Köln 2005, URL: <http://www.cei.lmu.de/pub/MagArbAssmann.pdf>.
- BACKMANN, Reinhold: Die Gestaltung des Apparates in den kritischen Ausgaben neuerer deutscher Dichter, in: Euphorion 25 1924, S. 629–662.
- BANSKI, Piotr: Why TEI stand-off annotation doesn't quite work: and why you might want to use it nevertheless, in: Balisage Series on Markup Technologies 5 2010, URL: <http://www.balisage.net/Proceedings/vol5/html/Banski01/BalisageVol5-Banski01.html>.
- BART, Patricia R. Bart: Experimental markup in a TEI-conformant setting, in: Digital Medievalist 1 2006, URL: <http://www.digitalmedievalist.org/journal/2.1/bart/>.
- BECHER, Margit: XML – DTD, XML-Schema, XPath, XQuery, XSLT, XSL-FO, SAX, DOM, 2009.
- BEIN, Thomas: Textkritik. Eine Einführung in Grundlagen germanistisch-mediävistischer Editionswissenschaft, Frankfurt am Main (u.a.) 2007.
- BESCH (HRSG), Werner: Sprachgeschichte. Ein Handbuch zur Geschichte der deutschen Sprache und ihrer Erforschung. 2. vollständig neu bearbeitete und erweiterte Auflage, Berlin 1998.
- BIRD, Steven und Mark LIBERMAN: A Formal Framework for Linguistic Annotation, in: Linguistic Data Consortium 2000, URL: <http://arxiv.org/abs/cs/0010033>.
- BOHNENKAMP, Anne: Neugermanistische Editionswissenschaft, in: Kompendium der Editionswissenschaften, URL: http://www.edkomp.uni-muenchen.de/CD1/frame_edkomp_AB.html.
- BOUBAKER, Houcine, Monji KHERALLHA und Adel M. ALIM: New Algorithm of Straight or Curved Baseline Detection for Short Arabic Handwritten writing, in:

- 10th International Conference on Document Analysis and Recognition 2009, S. 778–781.
- BOWMAN, Alan K. u. a.: A Virtual Research Environment for the Study of Documents and Manuscripts, in: *Digital Research in the Study of Classical Antiquity* 2010, S. 87–103.
- BOYDSTON, Jo Ann: In Praise of Apparatus, in: *Text* 5 1991, S. 1–13.
- BRADLEY, John: Tools to augment scholarly activity: an architecture to support text analysis, in: *Augmenting Comprehension: Digital Tools and the History of Ideas* 2004, hrsg. v. D. BUZZETTI, G. PANCALDI und H. SHORT, S. 19–48.
- BRÜNING, Gerrit, Katrin HENZEL und Dietmar PRAVIDA: Multiple Encoding in Genetic Editions: The Case of 'Faust', in: *Journal of the Text Encoding Initiative* 2013, URL: <http://jtei.revues.org/697>.
- BUBENHOFER, Noah u. a.: XML-Technologien als Grundlage dynamischer Textpräsentation. Die digitale Quellenedition Der Zürcher Sommer 1968, in: *Jahrbuch für Computerphilologie* 9 2009, URL: <http://computerphilologie.digital-humanities.de/jg07/bubenh.html>.
- BUBROWSKI, Ulrich: Editorische Schach- und Winkelzüge oder Versuch, unbeherrschbarer Schreibverhältnisse Herr zu werden - am Beispiel Barlach, in: *Editio. Beihefte*, Teil 10 1998, S. 117–154.
- BURNARD, Lou: From two cultures to digital culture: the rise of the digital demotic, in: *Informatica Umanistica. Dalla ricerca all'insegnamento* 2003, hrsg. v. Domenico FIORMONTE, URL: <http://users.ox.ac.uk/~lou/wip/twocults.html>.
- BUZZETTI, Dino: Diacritical ambiguity and markup, in: *Augmenting Comprehension: Digital Tools and the History of Ideas* 2004, hrsg. v. D. BUZZETTI, G. PANCALDI und Short H., S. 175–188.
- DERS.: Digital Editions: Variant Readings and Interpretations, in: *ALLC-ACH 96 Conference Abstracts* 1996, S. 254–256, URL: <http://gandalf.aksis.uib.no/allc-ach96/Panels/Thaller/buzzetti.html>.
- BUZZETTI, Dino und Andrea TABARRONI: Database Edition of Non-collatable Textual Traditions. In: *The Electric Scriptorium: Electronic Approaches to the Imaging, Transcription, Editing, and Analysis of Medieval Manuscript Texts* 1995.
- DERS.: Informatica e critica del testo: il caso di una tradizione 'fluida', in: *Schede umanistiche*, N.S., I 2 1991, S. 185–193.
- CASEY, R.G. und E. LECOLINET: Strategies in character segmentation: A survey, in: *3rd International Conference on Document Analysis and Recognition* 1995, S. 1028–1033.
- CAYLESS, Hugh A.: Linking Text and Image with SVG, in: *Kodikologie und Paläographie im digitalen Zeitalter - Codicology and Palaeography in the Digital Age*.

- Schriften des Instituts für Dokumentologie und Editorik 2 (2009), S. 145–158, URL: <http://kups.ub.uni-koeln.de/2967/>.
- CLOPPET, Florence: New Tools for Exploring, Analysing and Categorising Medieval Scripts, in: Digital Medievalist 7 2011, URL: <http://www.digitalmedievalist.org/journal/7/cloppet/>.
- CRAIG-McFEELY, Julia: Digital Image Archive of Medieval Music: The evolution of a digital resource, in: Digital Medievalist 3 2008, URL: <http://www.digitalmedievalist.org/journal/3/mcfeely/>.
- CUMMINGS, James: A Companion to Digital Literary Studies, (Part IV Methodologies: The Text Encoding Initiative and the Study of Literature), hrsg. v. Susan SCHREIBMAN und Ray SIEMENS, Oxford 2008, URL: <http://www.digitalhumanities.org/companionDLS/>.
- DERS.: Converting Saint Paul: A new TEI P5 edition of The Conversion of Saint Paul using stand-off methodology, in: Literary and Linguistic Computing (24.3) 2009, S. 307–317.
- CZMIEL, Alexander: Adäquate Markupsysteme für die digitale Behandlung altägyptischer Texte (Magisterarbeit), Universität zu Köln 2003, URL: <http://old.hki.uni-koeln.de/studium/MA/>.
- DERS.: Editio ex machina - Digital Scholarly Editions out of the Box, in: Digital Humanities 08 2008, S. 101–102, URL: www.ekl.oulu.fi/dh2008/Digital%20Humanities%202008%20Book%20of%20Abstracts.pdf.
- DAHLSTRÖM, Mats: Drowning by Versions, in: Human IT. Tidskrift för studier av IT ur ett humanvetenskapligt perspektiv 2000, URL: <http://etjanst.hb.se/bhs/ith/4-00/md.htm>.
- DAILEY, David: An SVG Primer for Today's Browsers, in: W3C Editor's Draft 2010, URL: <http://www.w3.org/Graphics/SVG/IG/resources/svgprimer.html>.
- DEUTSCHLAND, Nuntiaturberichte aus: III. Abteilung: 1572-1585, 7. Bd.: Nuntiatur Giovanni Dolfin (1573-1574), bearb. im Auftrag des Deutschen Historischen Instituts in Rom von Almut Bues, Tübingen 1990.
- D.P. O'DONNELL C. Karkov, J. Graham W. Osborn und R. Rosselli Del TURCO: The Visionary Cross: An Experiment in the Multimedia Edition. (Conference Paper), in: Proceedings of Digital Humanities 2007.
- ERBACHER, C.: Unser Denken bleibt gefragt: Web 3.0 und Wittgensteins Nachlass, in: Wissenschaftstheorie, Sprachkritik und Wittgenstein 2011, hrsg. v. S. WINDHOLZ und W. FEIGL, S. 135–146.
- FANTA, Walter: Die Entstehungsgeschichte des 'Mann ohne Eigenschaften' von Robert Musil, Wien 2000.

- FANTA, Walter: Die Klagenfurter Ausgabe Robert Musil. Historisch-kritisches Edieren am Computer, in: Jahrbuch für Computerphilologie 8 2007, URL: <http://computerphilologie.digital-humanities.de/jg06/fanta.html>.
- DERS.: Robert Musil – Klagenfurter Ausgabe. Eine historisch-kritische Edition auf DVD, in: Editio 24 2010, S. 117–148.
- DERS.: Zur Immortalität elektronischer Korpora am Beispiel der Musil-Edition, in: Jahrbuch für Computerphilologie (11 noch unveröffentlicht) 2011, URL: <http://computerphilologie.digital-humanities.de/jg09/fanta.html>.
- FEINEIS, Markus: Wortgenaue Annotation digitalisierter mittelalterlicher Handschriften, Bayerische Julius-Maximilians-Universität Würzburg. Institut für Informatik 2008.
- FENTON, Eileen Gifford und Hoyt N DUGGAN: Effective Methods of Producing Machine-Readable Text from Manuscript and Print Sources, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD, Katherine O'BRIEN O'KEEFFE und John UNSWORTH, S. 241–254, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/duggan.xml.
- FIORMONTE, D. und C. PUSCEDDU: The Text as Product and Process. History, Genesis, Experiments, in: Manuscript, Variant, Genese – Genesis 2006, hrsg. v. E. VANHOUTTE, S. 109–128.
- FIORMONTE Domenico Martiradonna, Valentina Schmidt Desmond: Digital Encoding as a Hermeneutic and Semiotic Act: The Case of Valerio Magrelli, in: Digital humanities quaterly (4.1) 2010, URL: <http://www.digitalhumanities.org/dhq/vol/4/1/000082/000082.html>.
- G. KEATING, John u. a.: A Digital Edition of a Spanish 18th Century Account Book: Part 1 – User Driven Digitisation, in: Jahrbuch für Computerphilologie 5 1010, URL: <http://computerphilologie.digital-humanities.de/jg08/keating1.html>.
- GABLER, Hans Walter: Das wissenschaftliche Edieren als Funktion der Dokumente, in: Jahrbuch für Computerphilologie 8 2007, URL: <http://computerphilologie.digital-humanities.de/jg06/gabler.html>.
- DERS.: For Ulysses – A Once and a Future Edition, in: Variants – The Journal of the European Society for Textual Scholarship 1 2002, S. 85–105.
- GAU, Melanie und Robert SABLATNIG: Technical Approaches To Manuscript Analysis and Reconstruction, in: Universität Wien 2007, S. 1–41, URL: http://www.zde.uni-wuerzburg.de/fileadmin/04100700/_temp_/GauSabltnig.pdf.
- GAU, Melanie u. a.: Image Acquisition and Processing Routines for Damaged Manuscripts, in: Digital Medievalist 6 2010, URL: <http://www.digitalmedievalist.org/journal/6/gau/>.

- GERIKE, Inga: Der Wanderer und sein Schatten. Manuskripte und Genese, ohne Datum, URL: <http://www.hypernetzsche.org/events/lmu/gerike-1.html>.
- GIBBS, Fred: New Textual Traditions from Community Transcription, in: Digital Medievalist 7 2011, URL: <http://www.digitalmedievalist.org/journal/7/gibbs/>.
- GISI, Lucas Marco, Hubert THÜRING und Irmgard M. WIRTZ: Schreiben und Streichen. Zu einem Moment produktiver Negativität. Tagungsband, hrsg. v. Irmgard M. WIRTZ, Göttingen, Zürich 2011.
- GOEDEKE (HRSG.), Karl: Schillers sämtliche Schriften. Historisch-kritische Ausgabe, Bd. 2, Stuttgart 1876.
- GREENSTEIN, Daniel: Conceptual Models and Model Solutions: A Summary Report of the TEI's Working Group on Historical Studies, in: Modelling Historical Data: Towards a Standard for Encoding and Exchanging Machine-Readable Texts. (A.11) 1991, S. 195–204.
- GREENSTEIN, Daniel und Lou BURNARD: Speaking with One Voice – Encoding Standards and the Prospects for an Integrated Approach to Computing in History, in: Computers and the Humanities 29.2 (1995), S. 137–148.
- GRÉSILLON, Almuth: Bemerkungen zur französischen 'édition génétique', in: Editio. Beihefte, Teil 10 1998, S. 52–64.
- GRÉSILLON, Almuth und Frauke ROTHER: Literarische Handschriften. Einführung in die 'critique génétique', Bd. 4 (Arbeiten zur Editionswissenschaft), Bern u.a. 1998.
- GUILLAUME, Joutel u. a.: Curvelets based Feature Extraction of handwritten shapes for ancient manuscripts classification, in: Laboratoire d'InfoRmatique en Image et Systèmes d'information 2007, S. 1–12, URL: liris.cnrs.fr/Documents/Liris-2688.pdf.
- GÖTTSCHE, Dirk: Ausgabentypen und Ausgabenbenutzer, in: Text und Edition. Positionen und Perspektiven 2000, hrsg. v. Rüdiger NUTT-KOFOTH u. a., S. 37–64.
- GÖTTSCHE, Dirk und Monika ALBRECHT: Ingeborg Bachmanns 'Todesarten'-Projekt. Elektronische Vorbereitung und Erschließung der kritischen Edition, in: Historical Social Research / Historische Sozialforschung (21.3) 1996, S. 154–160.
- HAAF, Susanne, Frank WIEGAND und Alexander GEYKEN: Measuring the Correctness of Double-Keying: Error Classification and Quality Control in a Large Corpus of TEI-Annotated Historical Text, in: Journal of the Text Encoding Initiative 2013, URL: <http://jtei.revues.org/739>.
- HANNA, Ralph: The application of thought to textual criticism in all modes - with apologies to A. E. Housman, in: Studies in Bibliography 53 2000, S. 163–172.
- HANSEN, Thomas: TEI - Keeping It Simple, in: Digital Medievalist 7 2011, URL: <http://www.digitalmedievalist.org/journal/7/hansen/>.

- HAUTZINGER, Nina: Vom Buch zum Internet? Eine Analyse der Auswirkungen hypertextueller Strukturen auf Text und Literatur, St. Ingbert 1999.
- HAYLES, N. Katherine: Translating media - Why we should rethink textuality, in: *Yale Journal of Criticism* (16.2) 2003, S. 263–290.
- HEROLD, Norbert und Sibylle MISCHER (HRSG.): Philosophie. Studium, Text und Argument, Münster 2003.
- HINMAN, Charlton: Mechanized Collation at the Houghton Library, in: *Library Bulletin* 9 1955, S. 132–134.
- HOCKEY, Susan M.: *Electronic Texts in the Humanities: Principles and Practice*, Oxford 2000.
- HOFFMANN, Hartmut: Die Edition in den Anfängen der Monumenta Germaniae Historica, in: *Mittelalterliche Texte. Überlieferung – Befunde – Deutungen. Kolloquium der Zentralkommission der MGH am 28./29. Juni 1996*, (MGH Schr. 42) 1996, hrsg. v. Rudolf SCHIEFFER, S. 189–232.
- HOWE, C. J. und Peter M. W. ROBINSON: The Phylogeny of The Canterbury Tales, in: *Nature* 394, URL: www.canterburytalesproject.org/pubs/nature.pdf.
- HUITFELDT, Claus: Editorial principles of Wittgenstein's Nachlass the Bergen electronic edition, in: *Augmenting Comprehension: Digital Tools and the History of Ideas* 2004, hrsg. v. D. BUZZETTI, G. PANCALDI und Short H., S. 113–128.
- DERS.: MECS (1992). Die BEE ist Wittgenstein's Nachlass. The Bergen Electronic Edition, Oxford 1998.
- DERS.: Multi-Dimensional Texts in a One-Dimensional Medium, in: *Computers and the Humanities* (28 4.5) 1995, S. 235–241.
- DERS.: Philosophy Case Study, in: *Electronic Textual Editing* 2006, hrsg. v. Lou BURNARD, Katherine O'BRIEN O'KEEFFE und John UNSWORTH, S. 181–196.
- HULLE, Dirk van: Authenticity or Hyperreality in Hypertext Editions. Notes Towards a Searchable 'Recherche', in: *Human IT 1. Tidskrift för studier av IT ur ett humanvetenskapligt perspektiv* 1999, URL: etjanst.hb.se/bhs/ith/1-99/dvh.htm.
- DERS.: 'Erronymous' Intentions from Joyce to Danielewski, in: *Variants – The Journal of the European Society for Textual Scholarship* 1 2002, S. 123–141.
- HUMBERTUS, de Romanis: *Exempla medii aevi*, hrsg. v. Christine [Hrsg.] BOYER, Bd. 4: *Humberti de Romanis De dono timoris (Corpus Christianorum : Continuatio mediaevalis; 218)*, Turnhout 2008.
- HURLEBUSCH, Klaus: Den Autor besser verstehen: aus seiner Arbeitsweise. Prolegomenon zu einer Hermeneutik textgenetischen Schreibens, in: *Textgenetische Edition* 1998, hrsg. v. Hans ZELLER, S. 7–51.

- HURLEBUSCH (HRSG.), Klaus: Friedrich Gottlieb Klopstock: Werke und Briefe. Historisch-kritische Ausgabe. Band 2, Berlin, New York 1977.
- JENKS, Stuart: KISS [Keep It Simple, Stupid]: Elektronische Quelleneditionen mit einfachsten Mitteln, in: Quellen und Quelleneditionen im neuen Medienzeitalter 2002, hrsg. v. Manfred THALLER, S. 27–37.
- KAMZELAK, Roland S.: Hypermedia – Brauchen wir eine neue Editionswissenschaft?, in: Beihefte zu Editio, Bd 12: Computergestützte Text-Editionen 1999, S. 119–126.
- DERS.: The Hybrid-Edition of Harry Count Kessler's Diary, in: Perspectives of Scholarly Editing / Perspektiven der Textedition. Hg. von H.T.M. van Vliet und Bodo Plachta. 2002, S. 181–190.
- DERS.: Zur Nachhaltigkeit von elektronischen Texten: XML und TEI, in: Was ist Textkritik? Zur Geschichte und Relevanz eines Zentralbegriffs der Editionswissenschaft 2009, S. 3–19.
- KANZOG, Klaus: Einführung in die Editionsphilologie der neueren deutschen Literatur (Grundlagen der Germanistik, Teil 31), Berlin 1991.
- KARASCH, Bernt: Critical Edition Typesetter (CET). Ein Programm zum Satz textkritischer Editionen auf PCs, in: Beihefte zu Editio, Bd 12: Computergestützte Text-Editionen 1999, S. 87–99.
- KARLSSON, Lina und Linda MALM: Revolution or Remediation? A Study of Electronic Scholarly Editions on the Web, in: Human IT (7.1) 2004, S. 1–46.
- KIERNAN, Kevin: Digital Faksimiles in Editing, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD, Katherine O'BRIEN O'KEEFFE und John UNSWORTH, S. 262–268, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/kiernan.xml.
- KILLY, Walther und Hans SZKLENAR (HRSG.): Georg Trakl: Dichtungen und Briefe. Band 2, Salzburg 1969.
- KLINE, Mary-Jo: A Guide to Documentary Editing, Baltimore 1987.
- KRAFT, Herbert: Editionsphilologie, Darmstadt 1990.
- LAVAGNINO, John: When not to use TEI, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD, Katherine O'BRIEN O'KEEFFE und John UNSWORTH, S. 334–338, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/lavagnino.xml?style=printable.
- LEBRAVE, Jean-Louis: Genetische Textkritik und Edition in Frankreich. Beiträge zur VII. Internationalen Fachtagung der Arbeitsgemeinschaft philosophischer Editionen (12.-14. März München), in: Editio. Beihefte, Teil 11 1998, S. 113–122.

- LEBRAVE, Jean-Louis: Hypertexts - Memories - Writing, in: *Genetic Criticism. Texts and Avant-textes* 2004, hrsg. v. Jed DEPPMAN, Daniel FERRER und Michael GRODEN, S. 218–237.
- LEBRAVE, Jean-Louis: Hypertext und textgenetische Edition, in: *Beihefte zu Editio*, Bd 10: *Textgenetische Edition* 1998, hrsg. v. Hans ZELLER, S. 329–345.
- LECOLINET, Eric, Laurent ROBERT und Francois ROLE: Text-image Coupling for Editing Literary Sources, in: *Computers and the Humanities* 36 2002, S. 49–73.
- LERNOUT, Geert: 'Critique génétique' und Philologie, hrsg. v. Rüdiger NUTT-KOFOTH, 2000, S. 121–142.
- LUEHRS, Kai: Verwirklichung oder Entzweiung? Zur Edition des Musil-Nachlasses auf CD-ROM, in: *Editio* 8 1994, S. 158–172.
- MARTENS, Gunter: Texterschließung durch Edition. Überlegungen zur rezeptionsästhetischen Bedeutung textgenetischer Apparate, in: *LiLi* (19.20) 1975, S. 82–104.
- MEYER, Bruno: Zur Edition historischer Texte, in: *Schweizerische Zeitschrift für Geschichte* (SZG 1) 1951, S. 177–202.
- MONELLA, Paolo: Towards a digital model to edit the different paratextuality levels within a textual tradition, in: *Digital Medievalist* 4 2008, URL: <http://www.digitalmedievalist.org/journal/4/monella/>.
- MORBIDONI, C. und M. NUCCI: SwickyNotes user guide, 2010, S. 237–246, URL: www.swickynotes.org/docs/SWickyNotesStartingGuide.pdf.
- NUTT-KOFOTH, Rüdiger: Edition und Interpretation Moderner Lyrik seit Hölderlin. In: *Beihefte zu Editio* 2010, hrsg. v. Dieter BURDORF, S. 17–39.
- DERS.: Schreiben und Lesen: Für eine produktions- und rezeptionsorientierte Präsentation des Werktextes in der Edition, in: *Text und Edition. Positionen und Perspektiven* 2000, hrsg. v. Rüdiger NUTT-KOFOTH u. a., S. 203–229.
- NÜNNING, Ansgar: Ansätze - Personen - Grundbegriffe, in: *Metzler Lexikon Literatur- und Kulturtheorie* 2004.
- O'HARA, Robert und Peter ROBINSON: Computer-assisted methods of stemmatic analysis. *Canterbury Tales Project*, in: *Occasional Papers I* 1993, S. 53–74, URL: www.canterburytalesproject.org/pubs/op1-cladistics.pdf.
- ORE, Espen S.: Document Markup – Why? How?, in: *Historical Social Research / Historische Sozialforschung* (37.3) 2012, S. 106–124.
- ORE, Espen S. und P. CRIPPS: The Electronic Publication of Wittgenstein's Nachlaß, in: *The Digital Demotic. DRH97 Conference, Digital Resources for the Humanities* 1998, hrsg. v. Lou BURNARD, Marilyn DEEGAN und Harold SHORT, S. 111–118.
- OTT, Tobias: Datenaufbereitung für elektronische Publikationen, in: *Beihefte zu Editio*, Bd 12: *Computergestützte Text-Editionen* 1999, S. 79–85.

- OTT, Wilhelm, Hans Walter GABLER und Paul SAPPLER: EDV-Fibel für Editoren, Tübingen 1982.
- (O.V.), Cisco Systems Inc.: Global Mobile Data Traffic Forecast Update, 2010-2015, in: Cisco Visual Networking Index 2011, URL: newsroom.cisco.com/dlls/ekits/Cisco_VNI_Global_Mobile_Data_Traffic_Forecast_2010_2015.pdf.
- PICHLER, A und A. ZÖLLNER-WEBER: Towards Wittgenstein on the Semantic Web, in: Chaucer Review (38.2) 2012, S. 318–321, URL: <http://www.dh2012.uni-hamburg.de/conference/programme/abstracts/towards-wittgenstein-on-the-semantic-web/>.
- PICHLER, Alois: Advantages of a Machine-Readable Version of Wittgenstein's Nachlass, in: Culture and Value. Beiträge des 18. Internationalen Wittgenstein Symposiums 1995, hrsg. v. Johannessen KJELL S. und Tore NORDENSTAM, S. 770–776, URL: wab.uib.no//alois/pichler-kirchb95b.pdf.
- DERS.: Encoding Wittgenstein. Some remarks on Wittgenstein's Nachlass, the Bergen Electronic Edition, and future electronic publishing and networking, in: TRANS. Internet-Zeitschrift für Kulturwissenschaften 10 2002, URL: <http://www.inst.at/trans/10Nr/pichler10.htm>.
- DERS.: Transcriptions, Texts and Interpretation, in: Culture and Value. Beiträge des 18. Internationalen Wittgenstein Symposiums 1995, hrsg. v. Kjell S. JOHANNESSEN und Tore NORDENSTAM, S. 690–695, URL: wab.uib.no//alois/pichler-kirchb95a.pdf.
- DERS.: Wittgensteins Philosophische Untersuchungen: Vom Buch zum Album, in: Studien zur Österreichischen Philosophie 36 2004, hrsg. v. R. HALLER.
- PIERAZZO, Elena: The Encoding of Time in Manuscript Transcription: Toward Genetic Digital Editions, in: Digital Humanities 2007, Conference Abstracts 2007, S. 150–152, URL: <http://www.ideals.uiuc.edu/handle/2142/2368>.
- PIEZ, Wendell: LMNL in Miniature. An introduction, in: Amsterdam Goddag Workshop 2008, URL: <http://piez.org/wendell/LMNL/Amsterdam2008/presentation-slides.html>.
- PLACHTA, Bodo: Editionswissenschaft. Eine Einführung in Methode und Praxis der Edition neuerer Texte, 2. ergänzte und aktualisierte Auflage, Ditzingen 2006.
- POLHEIM, Karl: Text und Textgeschichte des 'Taugenichts'. Eichendorffs Novelle von der Entstehung bis zum Ende der Schutzfrist, Tübingen 1989.
- POROMBKA, Stephan: Literaturbetriebskunde. Zur 'genetischen Kritik' kollektiver Kreativität, in: Kollektive Kreativität (Jahrbuch für Kulturwissenschaften und ästhetische Praxis) 2006, hrsg. v. Stephan POROMBKA, Wolfgang SCHNEIDER und Volker WORTMANN, S. 72–87.
- PRICE, Kenneth M.: Electronic Scholarly Editions. (Teil IV Methodologies: The Text Encoding Initiative and the Study of Literature), hrsg. v. Susan SCHREIB-

- MAN und Ray SIEMENS, Oxford 2008, URL: <http://www.digitalhumanities.org/companionDLS/>.
- PRÄTOR, Klaus: Ceci n'est pas un texte? Zur Rede über die Materialität von Texten – insbesondere in den Zeiten ihrer Digitalisierung. In: Materialität in der Editions-wissenschaft 2010, hrsg. v. Martin SCHUBERT, S. 421–428.
- REHBEIN, Malte: The transition from classical to digital thinking. Reflections on Tim McLoughlin, James Barry and collaborative work, in: Jahrbuch für Computerphilologie 10 2008, URL: computerphilologie.digital-humanities.de/jg08/rehbein.pdf.
- RENEAR, Allan, Elli MYLONAS und David DURAND: Refining our Notion of What Text Really Is: The Problem of Overlapping Hierarchies, in: Text Encoding Initiative: Background and Contexts 1993, URL: <http://www.stg.brown.edu/resources/stg/monographs/ohco.html>.
- RENEAR, Allen: Literal Transcription – Can the text ontologist help?, in: New Media and the Humanities: Research and Applications 2001, hrsg. v. Domenico FIORMONTE und Jonathan USHER, S. 23–30.
- DERS.: Out of Praxis: Three (Meta) Theories of Textuality, in: Electronic Text 1997, hrsg. v. K. SUTHERLAND, S. 107–126.
- RICKLEFS, Ulfert: Zur Systematik historisch-kritischer Ausgaben, in: Editio 13 1999, S. 1–22.
- RITTER, Jörg, Susanne SCHÜTZ und Stefan TEITGE: Entwicklung und Einsatz einer TEI-konformen Arbeitsumgebung für die Edition der Dramen von Karl Ferdinand Gutzkow, in: Jahrbuch für Computerphilologie 9 2007, URL: <http://computerphilologie.digital-humanities.de/jg07/rischuetei.html>.
- ROBERTSON, Paul und Robert LADDAGA: The GRAVA Self-Adaptive Architecture: History, Design, Applications and Challenges, in: Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference 2004, S. 298–303, URL: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.2.1328&rep=rep1&type=pdf>.
- ROBINSON, Peter: ... but what kind of electronic editions should we be making?, in: ALLC-ACH 96 Conference Abstracts 1996, S. 81–82, URL: <http://helmer.aksis.uib.no/allc/finneran.pdf>.
- DERS.: The Canterbury Tales and other Medieval Texts, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD, Katherine O'BRIEN O'KEEFFE und John UNSWORTH, S. 74–91, URL: http://www.tei-c.org/About/Archive_new/ETE/Preview/robinson.xml.
- DERS.: The History, Discoveries and Aims of the Canterbury Tales Project, in: Chaucer Review (38.2), S. 126–139, URL: <http://www.canterburytalesproject.org/pubs/PR-ChauRev.pdf>.

- DERS.: What is an Electronic Critical Edition?, in: *Variants: The Journal of the European Society for Textual Scholarship* 1 2002, S. 51–57.
- DERS.: Where We Are with Electronic Scholarly Editions, and Where We Want to Be, in: *Jahrbuch für Computerphilologie* 5 2004, URL: <http://computerphilologie.digital-humanities.de/jg03/robinson.html>.
- ROBINSON, Peter und Kevin TAYLOR: Publishing an Electronic Textual Edition: The Case of The Wife of Bath's Prologue on CD-ROM, in: *Computers and the Humanities* (32.4) 1998, S. 271–284.
- ROMARY, Laurent: Questions and Answers for TEI Newcomers, in: *Jahrbuch für Computerphilologie* 10 2008, URL: <http://computerphilologie.digital-humanities.de/jg08/romary.pdf>.
- ROSS, Charles L.: A future for Editing: Lawrence in Hypertext, in: *Textual studies and the common reader: essays on editing novels and novelists* 2000, S. 141–159.
- ROUED-CUNLIFFE, Henriette: Towards an Interpretation Support System for Reading Ancient Documents, in: *Literary and Linguistic Computing. Centre for the Study of Ancient Documents* 2010, S. 1–15, URL: http://www.academia.edu/1786977/Towards_an_Interpretation_Support_System_for_Reading_Ancient_Documents.
- SAHLE, Patrick: *Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels*, Bd. 1 Das typographische Erbe, Norderstedt 2013.
- DERS.: *Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels*, Bd. 2 Befunde, Theorie und Methodik, Norderstedt 2013.
- DERS.: *Digitale Editionsformen. Zum Umgang mit der Überlieferung unter den Bedingungen des Medienwandels*, Bd. 3 Textbegriffe und Recodierung, Norderstedt 2013.
- SALLER, Harald: HNML – HyperNietzsche Markup Language, in: *Jahrbuch für Computerphilologie* 5 2003, S. 183–190, URL: <http://computerphilologie.digital-humanities.de/jg03/saller.html>.
- SANZ, Amelia und Maria GOICOECHEA: Literary reading rituals and practices on new interfaces, in: *Journal of the Alliance of Digital Humanities Organizations*. (27.3) 2012.
- SCHEIBE, Siegfried: Welche Editionsart für welchen Zweck?, in: *Berliner Beiträge zur Editionswissenschaft* 3 (1998), S. 43–61.
- SCHMIDT, Desmond: Merging Multi-Version Texts: a General Solution to the Overlap Problem, in: *Balisage Series on Markup Technologies* 3 2009, URL: <http://www.balisage.net/Proceedings/vol3/html/Schmidt01/BalisageVol3-Schmidt01.html>.

- SCHMIDT, Desmond: The Role of Markup in the Digital Humanities, in: *Historical Social Research / Historische Sozialforschung* (37.3) 2012, S. 125–146.
- SCHMIDT, Desmond, Nicoletta BROCCA und Domenico FIORMONTE: A Multi-version Wiki, in: *Digital Humanities 08 2008*, S. 187–188, URL: www.ekl.oulu.fi/dh2008/Digital%20Humanities%202008%20Book%20of%20Abstracts.pdf.
- SCHREIBMAN, Susan: The Text Encoding Initiative. An Interchange Format Once Again, in: *Jahrbuch für Computerphilologie* 10 2008, URL: <http://computerphilologie.digital-humanities.de/jg08/schreibman.html>.
- SCHULTE, Joachim: Wittgenstein's Nachlass: The Bergen Electronic Edition, in: *Grazer philosophische Studien* 65 2002, S. 237–246.
- SENIOR, W.: Off-line handwritten recognition: A review and experiments, in: *Technical Report CUED/F-INFENG/TR105* 1992, S. 1028–1033.
- SEUFFERT, Bernhard: Prolegomena zu einer Wieland-Ausgabe. III. IV. Berlin (= *Abhandlungen der preußischen Akademie der Wissenschaften*), 1905 (Reprint Hildesheim 1989).
- SHILLINGSBURG, Peter: Anglo-amerikanische Editionswissenschaft. Ein knapper Überblick, in: *Text und Edition – Positionen und Perspektiven* 2000, hrsg. v. Rüdiger NUTT-KOFOTH, Bodo PLACHTA und Hermann van VLIET H.T.M. und Zwerschina, S. 143–164.
- DERS.: Principles for Electronic Archives, Scholarly Editions, and Tutorials, in: *The Literary Text in the Digital Age*. Richard J. Finneran (Hrsg.) S. 23–35.
- DERS.: *Scholarly Editing in the Computer Age*. (3. Auflage), Ann Arbor 1996.
- DERS.: 'The subject of our Mirth'. The Aesthetic Object in Anglo-American Editing, in: *Perspectives of Scholarly Editing / Perspektiven der Textedition*. Hg. von H.T.M. van Vliet und Bodo Plachta. 2002, S. 97–112.
- SPERBERG-MCQUEEN, C.M., Lou BURNARD und Syd BAUMAN: TEI P5: Guidelines for Electronic Text Encoding and Interchange, Charlottesville 2013, URL: <http://www.tei-c.org/Vault/P5/2.0.1/doc/tei-p5-doc/en/Guidelines.pdf>.
- SPERBERG-MCQUEEN, C. M. und Claus HUITFELDT: Markup Discontinued: Discontinuity in TexMecs, Goddag structures, and rabbit/duck grammars, in: *Balisage Series on Markup Technologies* 1 2008, URL: <http://www.balisage.net/Proceedings/vol1/html/Sperberg-McQueen01/BalisageVol1-Sperberg-McQueen01.html>.
- SPERBERG-MCQUEEN, C.M. und Claus HUITFELDT: GODDAG. A Data Structure for Overlapping Hierarchies, in: *ALLC-ACH99* 1999.
- STEDING, Sören A.: Benutzerorientierte Digitale Editionen: Eine empirische Annäherung, in: www.germanistik2001.de. Vorträge des Erlanger Germanistentags 2002, hrsg. v. Hartmut KUGLER, S. 729–742.

- STOKES, Peter A.: Palaeography and Image-Processing: Some Solutions and Problems, in: Digital Medievalist 3 2008, URL: <http://www.digitalmedievalist.org/journal/3/stokes/>.
- STOLZ, Michael: Datenaufbereitung für elektronische Publikationen, in: Materialität in der Editionswissenschaft 2010, hrsg. v. DEMS., S. 275–291.
- TANSELLE, G. Thomas: Textual Instability and Editorial Idealism, in: Studies in Bibliography 49 1996, S. 1–60, URL: <http://etext.lib.virginia.edu/bsuva/sb/>.
- TEI: Workgroup on Genetic Editions. An Encoding Model for Genetic Editions, 2010, URL: <http://www.tei-c.org/Activities/Council/Working/tcw19.html>.
- TERRAS, Melissa M.: Image to Interpretation. An Intelligent System to Aid Historians in Reading the Vindolanda Texts, New York 2006.
- THALLER, Manfred: Controversies around the Digital Humanities: An Agenda, in: Historical Social Research / Historische Sozialforschung (37.3) 2012, S. 7–24.
- DERS.: Data Bases v. Critical Editions, in: Historical Social Research (13.3) 1988, S. 129–139.
- DERS.: Datenbasen als Editionsformen?, in: Historische Edition und Computer. Möglichkeiten und Probleme interdisziplinärer Textverarbeitung und Textbearbeitung 1989, hrsg. v. Anton SCHWOB, Karin KRANICH-HOFBAUER und Diethard SUNTINGER, S. 215–241.
- DERS.: Texts, Databases, Kleio: A Note on the Architecture of Computer Systems for the Humanities, in: Augmenting Comprehension: Digital Tools and the History of Ideas 2004, hrsg. v. D. BUZZETTI, G. PANCALDI und Short H., S. 49–76.
- THOMAS, William G.: Trends in Digital Humanities: remarks at the CIC Digital Humanities Summit ('the keynote in the dark'), 2012, URL: <http://railroads.unl.edu/blog/?p=794>.
- TUOMAS, Heikkilä und Teemu ROOS: Evaluating methods for computer-assisted stemmatology using artificial benchmark data sets, in: Literary and Linguistic Computing (24.4) 2009, S. 417–433.
- URCHUEGUIA, Cristina: Apparat, in: Kompendium der Editionswissenschaften, URL: http://www.edkomp.uni-muenchen.de/CD1/frame_edkomp_CU0.html.
- VANHOUTTE, Edward: Display or Argument: Markup and Visualisation for Electronic Scholarly Editions, in: Standards und Methoden der Volltextdigitalisierung. Beiträge des Internationalen Kolloquiums an der Universität Trier 2003, hrsg. v. Thomas BURCH u. a., S. 71–96.
- DERS.: Prose Fiction and Modern Manuscripts: Limitations and Possibilities of Text-Encoding for Electronic Editions, in: Electronic Textual Editing 2006, hrsg. v. Lou BURNARD, Katherine O'BRIEN O'KEEFFE und John UNSWORTH, S. 161–180.

- VANHOUTTE, Edward: Putting Time back in Manuscripts: Textual Study and Text Encoding with Examples from Modern Manuscripts, in: ALLC/ACH 2002 Abstracts 2002, S. 126–127, URL: <http://ctb.kantl.be/vanhoutte/pub/2002/allc02abstr.htm>.
- DERS.: Traditional editorial standards and the digital edition, in: Proceedings of the Emblem Project Utrecht Conference on Dutch Love Emblems and the Internet (November 2006) 2007, S. 157–174, URL: http://emblems.let.uu.nl/static/images/project/learned_love_157-174.pdf.
- DERS.: Where is the editor? Resistance in the creation of an electronic critical edition, in: Human IT. Tidskrift för studier av IT ur ett humanvetenskapligt perspektiv 1998, URL: <http://etjanst.hb.se/bhs/ith/1-99/ev.htm>.
- VIETOR, Sophia: Astralis von Novalis. Handschrift - Text - Werk (Stiftung für Romantikforschung, Teil 15), Würzburg 2001.
- VLIET, H. T. M. van: Electronic Editions. New Solutions for Old Editing Problems or Old Wine in New Bottles?, in: Variants: The Journal of the European Society for Textual Scholarship 1 2002, S. 63–83.
- VOSWINCKEL, Till: XML - SVG Presenter Strukturierte Multimedia-Präsentation im Web, Fachhochschule Furtwangen 2003, URL: www.carto.net/papers/till_voswinckel/paper_till_voswinckel_xml_svg.pdf.
- WIENERS, Jan Gerrit: Zur Erweiterungsfähigkeit bestehender OCR Verfahren auf den Bereich extrem früher Drucke (Magisterarbeit), Universität zu Köln 2008, URL: <http://old.hki.uni-koeln.de/studium/MA/>.
- WIESMÜLLER, Roland S.: Textkritik und Dekonstruktion. Überlegungen zu neuen textgenetischen Modellen anhand der Internet-Edition der 'Witiko'-Handschriften von Adalbert Stifter, in: Was ist Textkritik? Zur Geschichte und Relevanz eines Zentralbegriffs der Editionswissenschaft 2009, hrsg. v. Gertraud MITTERAUER, S. 283–290.
- WINDRAM, Heather F., Christopher J. HOWE und Matthew SPENCER: The identification of exemplar change in the Wife of Bath's Prologue using the maximum chi-squared method, in: Literary and Linguistic Computing (20.2) 2005, S. 189–204.
- WINDRAM, Heather F., Prue SHAW und Peter ROBINSON: Dante's Monarchia as a test case for the use of phylogenetic methods in stemmatic analysis, in: Literary and Linguistic Computing (23.4) 2008, S. 443–463.
- WISCHUSEN, Jeremy J.: Scalable Vector Graphics in HTML5. Basic concepts and usage, in: IBM developerWorks 2012, S. 1–22, URL: public.dhe.ibm.com/software/dw/web/wa-scalable/wa-scalable-pdf.pdf.

- ZELLER, Hans: Befund und Deutung. Interpretation und Dokumentation als Ziel und Methode der Edition, in: *Texte und Varianten* 1971, hrsg. v. Gunter MARTENS und Hans ZELLER, S. 45–90.
- DERS.: Die Typen des Germanistischen Variantenapparats und ein Vorschlag zu einem Apparat für Prosa, in: *Editionsprobleme der Literaturwissenschaft* 105, Sonderheft (1986), S. 42–69.
- DERS.: Fünzig Jahre neugermanistische Edition. Zur Geschichte und künftigen Aufgaben der Textologie, in: *Editio* 3 1989, S. 1–17.
- ZUNDERT, Joris van: If you Build it, Will We come? Large Scale Digital Infrastructures as a Dead End for Digital Humanities, in: *Historical Social Research / Historische Sozialforschung* (37.3) 2012, S. 106–124.
- ZWERSCHINA, Hermann: Variantenverzeichnung, Arbeitsweise des Autors und Darstellung der Textgenese, in: *Text und Edition. Positionen und Perspektiven* 2000, hrsg. v. Rüdiger NUTT-KOFOTH u. a., S. 203–231.
- DERS.: Von den Faksimiles zur Interpretation, in: *Perspectives of Scholarly Editing / Perspektiven der Textedition*. Hg. von H.T.M. van Vliet und Bodo Plachta. 2002.